



A Model-Based Approach to Battle Execution Monitoring

by Richard C. Kaste and Charles E. Hansen

ARL-TR-3394

January 2005

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

Army Research Laboratory

Aberdeen Proving Ground, MD 21005-5067

ARL-TR-3394

January 2005

A Model-Based Approach to Battle Execution Monitoring

Richard C. Kaste and Charles E. Hansen
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188		
<p>Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.</p> <p>PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.</p>					
1. REPORT DATE (DD-MM-YYYY) January 2005		2. REPORT TYPE Final		3. DATES COVERED (From - To) October 2002–September 2003	
4. TITLE AND SUBTITLE A Model-Based Approach to Battle Execution Monitoring			5a. CONTRACT NUMBER		
			5b. GRANT NUMBER		
			5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S) Richard C. Kaste and Charles E. Hansen			5d. PROJECT NUMBER FY03-CIS-04		
			5e. TASK NUMBER		
			5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) U.S. Army Research Laboratory ATTN: AMSRD-ARL-CI-CT Aberdeen Proving Ground, MD 21005-5067			8. PERFORMING ORGANIZATION REPORT NUMBER ARL-TR-3394		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)			10. SPONSOR/MONITOR'S ACRONYM(S)		
			11. SPONSOR/MONITOR'S REPORT NUMBER(S)		
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT <p>The work set forth in this report is intended to lead to portions of an integrated system to dynamically link automated plan generation and analysis with execution monitoring. The report deals with algorithms for, and interfaces to, a prototype battle-trajectory decision aid. We explored complexities of navigating parameter tradeoff spaces and performed detailed studies of strength/attrition parameter estimation and calibration techniques. We investigated software for solving differential systems and for visualizing and interacting with decision aid data. We considered aspects of the analytical theory of dynamic systems: mathematical chaos, stability, and control. We set the groundwork for formulation of models to produce systems exhibiting inherently chaotic behavior and for studying qualitative aspects of phase plane trajectories. It is probable that nonlinear analytic techniques will soon be considered as invaluable to commanders.</p>					
15. SUBJECT TERMS <p>combat modeling, military decision aiding, dynamical systems, cause of action analyses</p>					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UL	18. NUMBER OF PAGES 80	19a. NAME OF RESPONSIBLE PERSON Richard C. Kaste
a. REPORT UNCLASSIFIED	b. ABSTRACT UNCLASSIFIED	c. THIS PAGE UNCLASSIFIED			19b. TELEPHONE NUMBER (Include area code) 410-278-7781

Contents

Contents	iii
List of Figures	v
List of Tables	v
Acknowledgments	vi
1. Introduction	1
1.1 Project.....	1
1.2 Report.....	2
1.3 Background	3
2. Basic Idea	4
2.1 Concepts	4
2.2 Design.....	8
2.3 Decision Aid.....	9
3. Initial Formulation	10
3.1 Motivation	10
3.2 Model.....	11
3.3 Sketch of Algorithm	11
4. Theoretical Preliminaries	14
4.1 Fundamental Notions	14
4.2 Analytical Results.....	16
5. Aspects of the Decision Aid	20
5.1 Introduction	20
5.2 Trajectory Tracking.....	21
5.3 Sensitivity Analyses	23
5.4 Qualitative Behavior.....	26
5.5 Other Desiderata.....	27
6. Software	28
6.1 Consideration of Options.....	28

6.2	Dynamics Solver	30
6.3	Contingency.....	31
7.	Estimation of Parametric Values	33
7.1	Introduction	33
7.2	Recurrence Relation Derivation	35
7.3	Least Squares Approach.....	36
7.4	Modifications.....	37
7.5	Illustrations	39
8.	Techniques for Analysis of Dynamic Systems	45
8.1	Introduction	45
8.2	Theoretical Notions	46
8.3	Chaos	47
8.4	Stability	48
8.5	Deviation From Plan	51
8.6	Control Theory	52
9.	Future Efforts	53
9.1	Extension of Initial Model.....	53
9.2	Attrition	54
9.3	Next Formulation	55
9.4	Modeling Continuous Forces	58
9.5	Heuristics and Estimation.....	59
9.6	Other Applications	60
10.	Conclusion	62
10.1	Summary	62
10.2	Impact.....	63
10.3	Issues	64
10.4	Final Observations.....	64
11.	References	66
	Distribution List	70

List of Figures

Figure 1. An example of a battle trajectory.	5
Figure 2. Blue and Red force levels over time.....	13
Figure 3. Planned trajectory with alerts as actual trajectory breaks envelope.	23
Figure 4. GUI input of the square law solution.	32
Figure 5. Graph resulting from GUI input.	33
Figure 6. Predicted strength.	39
Figure 7. Smoothed estimate.....	40
Figure 8. Y-force strength levels.	41
Figure 9. Predicted strength levels.....	42
Figure 10. Modified situation.	42
Figure 11. Attrition rate quantities.....	43
Figure 12. Predicted Iraqi forces.....	44
Figure 13. Delayed casualty reports.....	44

List of Tables

Table 1. Concepts and symbols.	6
Table 2. Stability of (0,0) for $\{x' = ax + by, y' = cx + dy\}$	49

Acknowledgments

This work was performed under the auspices of the U.S. Army Research Laboratory (ARL) Director's Research Initiative FY03-CIS-04. The authors gratefully acknowledge the assistance of Janet O'May, Fred Brundick, Andy Neiderer, and Pat Jones in this project. Ms. O'May generated scenarios, implemented simulations, and wrote data collection/analysis programs. Mr. Brundick was instrumental in developing the improvements to the interface and plotting software discussed in subsection 6.3. Mr. Neiderer performed background research into interface software and facilitated the illustrations. Ms. Jones offered editorial suggestions that improved the readability of this report.

1. Introduction

1.1 Project

The project “A Model-Based Approach to Battle Execution Monitoring” involves a form of exploratory mathematics for military application that is intended to lead to more basic research with greater payoff. We compare an updated-parameter model trajectory to a planned-parameter model trajectory. There are interesting problems in doing such comparisons from a “least-squares” type standpoint and basins of attraction. As a practical matter, various tactical considerations such as attrition levels in selected time make such comparisons more straightforward. We examine such in this project in the hopes of being able to continue the more difficult research based on the tools and foundations developed here. This is not the innovative basic research, but rather sets the stage and tool formats for subsequent exploration and novelty. Briefings to advisory bodies have been generally positively received, and management has offered good suggestions for improvement.

This project was motivated somewhat by the changing nature of course of action (COA) analysis, in particular, incorporation of the notion of commander’s intent. This formed a justification for the gross modeling of battle in the abstract, which lends itself to qualitative analysis, use of rough estimates, and consideration of the derivatives of the trajectories of conflict. Here is a good place to mention a quote from the father of this sort of combat analysis, “There are many who will be inclined to object to any mathematical or semimathematical treatment of the present subject, on the ground that with so many unknown factors such as the morale or leadership of the men, the unaccounted merits or demerits of the weapons, and the still more unknown ‘chances of war,’ it is ridiculous to pretend to calculate anything. The answer to this is simple: the direct numerical comparison of the forces engaging in conflict or available in the event of war is almost universal. It is a factor always carefully reckoned with by the various military authorities; it is discussed *ad nauseum* in the press. Yet such direct counting of forces is, in itself, a tacit acceptance of the applicability of mathematical principles, but confined to a special case. To accept without reserve the mere ‘counting of the pieces’ as to value and to deny the more extended application of mathematical theory, is as illogical and unintelligent as to accept broadly and indiscriminately the balance and the weighing machine as instruments of precision, but to decline to permit in the latter case, any allowance for the known inequality of leverage (*I*).”

The usefulness of mathematical modeling for combat analysis can then be considered well-accepted. We can use it to understand somewhat the dynamics of conflict, and hence can invoke it, at least iteratively, as a tool for developing plans for the mission. But a more difficult problem, again one that helped motivate this project, is the question of what triggers the need for replanning; that is, how one realizes that the actual battle is deviating unacceptably from that

envisioned. This is certainly part of the military art of experienced commanders, but whether it can be brought practically into the realm of science remains an open question. One reason to monitor a battle is to see if it goes too far off track. How do we know it is off track?

The basic intent was to use this project as a launcher for novel research into other types of execution monitoring and dynamic replanning. An ongoing area of research is the problem of evaluation of “goodness” of a COA. A COA is a feasible way to accomplish a mission that follows the commander’s guidance, will not result in undue risk, and is noticeably different from other actions being considered (2). That is, what makes one plan “better” than another? Can it be better statically, as is often done by planners, or only dynamically? Can a plan be shown to be good only at the end of the battle?

We proposed an application, building on classic and newer techniques of mathematical modeling, to dynamic battlespace decision support. Technical challenges included investigations into goodness of the underlying model, determination of input parameters and values, and transformation of results for use in the tactical domain. We intend to develop a set of algorithms to be coded into a computerized prototype of a command staff decision aid. This work constitutes portions of a prototype system dynamically linking battle plan generation/analysis with execution monitoring. The technique should permit comparison of a plan, in terms of desired end state and other measures, with the trajectory of the conflict in phase space. During planning, and particularly during monitoring, if analysis shows the trajectory does not permit the end state, the user is to be helped with adjustments enabling the end state or be alerted to the best results achievable.

We sought application of results in nonlinear dynamics to such problems. There has been considerable philosophical discussion concerning the sensitivity of combat to small variations in initial conditions (“for lack of a nail the kingdom was lost”), but we seek mathematical rigor in treating combat as a dynamic system. We are also interested in whether notions of optimal control theory can be applied to such analyses, perhaps ultimately leading to tools for monitoring divergence from a battle plan and making changes. Can combat be analyzed (at least in a qualitative sense) that allows reasonable measurement of the influences of various factors?

1.2 Report

In this report, we concentrate on expositions concerning the main technical results with regard to the design and implementation of the prototype, including the nature of decision aids based on these concepts and particular parameter estimation techniques. These discussions are followed by subsections dealing with ancillary aspects (such as the interface and control theory) that will be developed as part of the follow-on work. Note that some of this is like a tutorial. This is on purpose. The mathematical derivation is necessary to enable the construction of algorithms and hence the prototype. We entered into this project with the intent, as explicitly laid out in the Director’s Research Initiative program, to justify further developments along these lines as part of the division mission.

Therefore, we set forth, in a kind of white paper, a summary of emerging results and areas we deem as having potential for fruitful further development. The format involves a series of discussions of the topics, each ranging from an introduction of concepts to more in-depth development and areas for further research. The reader will notice several aspects to developing a decision aid of this sort. We have made progress on several. We spell out various topics in the hope of coordinating efforts for future researchers, including interns and students. Various design desiderata are set forth as guidance for the coding of the software. Some additional discussion concerning the actual computer work is provided in later portions of the report. In particular, results on the parameter estimation routines and the curve plotting routines are examined in some detail.

We generally assume continuity, as opposed to discreteness, in the systems being discussed. However, in some instances discussion of discrete time intervals is more appropriate to the analysis, but this is clear in context and should cause the reader no particular confusion. A similar comment applies to nonlinearity, as opposed to linearity. Also, for the sake of brevity, in some cases the terminology is a bit loose, but this will be made stringent in future examinations of specific systems.

1.3 Background

A fundamental challenge of the 21st century battlefield is to reduce vast amounts of data into contextually relevant and actionable knowledge. Achieving the intent of the Chief of Staff's Army Vision to improve operational jointness and ensure responsive dominance will require cognitive amplifiers for application to Future Combat Systems. By ARL's treatment of combat modeling, we intend to facilitate provision of the right tools for evaluating and improving combat effectiveness. This research focuses on identifying and developing techniques for modeling and simulation to meet these fundamental challenges. The overall objective is to develop and evaluate COA tools for the mobile commander in the Future Force. Enhancement of the staff's planning capability will allow increased options in an engagement. The technical focus is on research into techniques for automated COA evaluation incorporating "reasonable-time" battlefield information and development of a COA analysis testbed and associated decision tools. An essential part of this work is extension of the mathematics of combat modeling.

Analysts use differential equations (DEs) to model aspects of combat. Parts of so-called Lanchester theory are fairly well developed. Such models, generally solved through computerized difference equations, can estimate the winner and duration of a conflict and track the history of force levels. More recent aspects include stochastic formulations and examination of optimal combat responses. Moreover, attrition of moving forces lends itself to use of partial DEs, which handle density and geometry, and these formulations may permit analysis of interpenetration and nonlinear battles in future scenarios.

Various techniques might be applied to the analysis/assessment/evaluation of a COA, given that it has already been derived. One of particular interest to analysts is wargaming, which may be

manual, mathematical, computerized, or a combination. Several approaches, mostly derived from wargaming, are traditionally somewhat feasible in examining changes to battlespace parameters (weapons and tactics). These include combat ratios, firepower potential, history, unit effectiveness, rate of advance, casualties, and vehicles lost. Innumerable ways exist to measure or calculate such things. Moreover, other methodologies utilize these battle statistics as inputs for subsequent numerical or symbolic analyses: decision trees, Bayesian belief nets, expert system advisors, etc. We would like to develop a procedure, automated if possible, for indicating the effectiveness of forces, COAs, and tactics in various scenarios. The intent is to utilize the killer-victim scoreboards (or other measures of effectiveness) extracted from computerized wargame outputs as inputs to the procedure. A theoretical problem is whether there could be a way to evaluate the general utility of forces (or other aspects of the battlespace) without regard to specific situations. What we basically seek is an optimal technique or (possibly combined) set of measures, probably "situation dependent," useful for evaluating COAs in general (3).

We considered many aspects of such a project before narrowing it down: the modularity of planning and gaming, ongoing work at ARL on genetic algorithms (4), improvement of combat simulations (5), commercial AI tools, military scenario-based games, display techniques, qualitative physics and model-based reasoning, training vice functional/operational applications, and so on. Some of the work to be discussed emerged from an attempt to address such notions.

2. Basic Idea

2.1 Concepts

We consider as a fundamental concept that one can compare plans or COAs within the context of a given scenario. (Many terms will take on technical meanings in the subsequent development.) Such comparison can be done, as alluded to earlier, statically; for instance, whether certain aspects of the envisioned conflict are addressed. However, it makes more sense to do the comparison in terms of "trajectory" and realized "end state," where trajectory and end state are related by notions to be discussed.

We can define a COA in terms of the paths it precipitates in a space of parameters describing the conflict. For example, a (rather simplistic) kind of Blue COA can be defined as a given rate of fire and a given rate of movement at a given time. Such a COA can then be tied into Lanchester modeling. Let us consider that a COA is analogous to a vehicle being controlled through an environment. The environment here reflects the hostile activities of the opponent: the Red "wind" blows the Blue position and strength back. We can track the Blue and Red strengths (or whatever attributes we can meaningfully assess) in another phase plot for monitoring. The details of such tracking, with regard to human-factors-based navigation of a parameter space,

will be discussed in section 5 on decision aid aspects. By way of introduction, figure 1 gives an example of a battle trajectory: the plot shows a schematic portrayal of the force ratio over time.

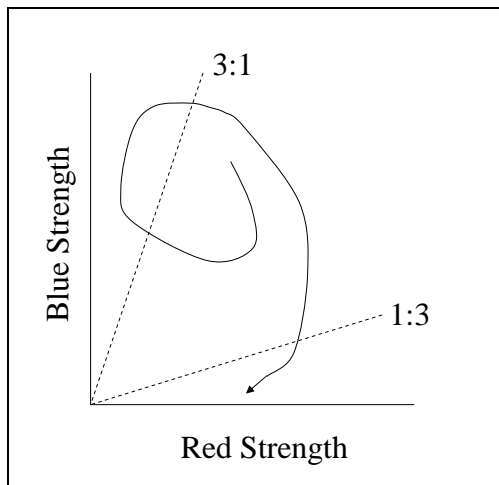


Figure 1. An example of a battle trajectory.

We therefore take it as a given that we can consider Blue and Red as having abstract trajectories in a “battlespace” with regard to missions, assets, and actions. The idea, then, is to consider a model of the conflict, say a system of DEs. The commander can control certain aspects of the battle. That is, he has reasonable power to modulate certain attributes. For instance, he might modify the number of his troops, via reinforcement or withdrawal, even in the midst of the conflict. To some extent, he can modify the rate at which he kills the enemy (say, increasing it by utilizing heavier firepower) and the rate at which the enemy inflicts attrition on his force (say, decreasing it by improving defensive posture). A desirable decision aid would constantly give the parametric set(s) that would enable attaining the goal. This could be demonstrated by having an (invisible) enemy changing his values.

Note there is really nothing special about the form of the model. As long as we have parameters that influence the trajectory, be it discrete or continuous or symbolic or whatever, we can proceed with such methodology. Indeed, as computational power increases, the nature of the model can be more and more sophisticated; as the Future Force becomes operational, the data to populate such models should become readily available in near real time. However, even very high fidelity models and accurate information may not approximate reality sufficiently well for the commander’s purposes. Therefore, another novel concept set forth in our work is that of attempting to adapt (or eventually even actually develop, probably through powerful statistical techniques) the model on the fly from observed data. A fundamental question, then, to be considered in some depth later in this report, concerns the feasibility (technical and tactical) of computing the parameters or the model itself from the observed values.

What does it mean for combat to be considered a dynamic system? As a lead in to the discussions of analytical methodologies of section 8, we note that a dynamic system can be

considered mathematically, in one sense, as a vector field. As a practical matter for initial development of the prototype decision aid, however, we will develop it in terms of set of differential equations describing aspects of behavior, basically as functions of time. Notions of space are, initially, fairly abstract.

The concepts and symbols set forth in table 1 will be necessary for the development of the theory and implementation of the proposed decision aid. Some of these are relatively straightforward, but most require some discussion of the underlying philosophy. (Note that no letter convention is followed with regard to a symbol denoting a constant or variable.)

Table 1. Concepts and symbols.

Concept	Symbol
Time (as variable)	t
Friendly (Blue) force level	x
Enemy (Red) force level	y
Rate at which Red kills Blue	a
Rate at which Blue kills Red	b
Blue breakpoint	p_x
Red breakpoint	p_y
Initial ($t = 0$)	t_0, x_0 (sub 0)
Specific time of interest	τ
Known/measured variable	x (no mark)
Desired by Blue	\tilde{x}_τ (tilde over variable)
Assumed by Blue	\bar{y}_τ (cup over variable)
Modeled (by Blue)	\hat{x}_τ (hat over variable)
(Blue) desired time to end	\tilde{t}_ω
End state (desired by Blue)	$\tilde{\Omega}$

“Breakpoint” connotes the percentage of its initial force level at which a force would attempt to disengage and break contact. The time to end desired by Blue represents the time that Red would be defeated, where this is of necessity in this scheme related to other Red parameter(s), in particular casualty level. “Known/measured” means that the Blue commander has sufficient ground truth or intelligence to take the value of the attribute as effectively true; “assumed,” on the other hand, means that the Blue commander is postulating this attribute to have a certain value for his purposes of assessing the mission; and “modeled” means that the value of this attribute has arisen from calculations of the active system of equations for the conflict. In a “degenerate” case, we can simplify the analysis somewhat by assuming the breakpoints are 0. That is, one side wins when the other is totally annihilated. However, this is not tactically realistic, and the algorithm is not overly complicated by dropping this assumption.

Kill rates (or attrition rates) require a little discussion. Anderson and Miercort (6) note that comparisons of combat forces are generally calculated as follows. All resources on each side are grouped into categories. Each resource is assigned a (nonnegative) score. These scores, which can be functions of the numbers and effectiveness parameters of the resources, are constant within categories and can vary across categories. Force strengths are formed by summing, over categories, the product of the number of resources in each category times the score. Comparisons of forces are then computed using (real-valued) functions of these strengths and, perhaps, other parameters. Thus one must consider how to calculate scores for resources, and how to combine them to form force strengths. We note that over the years much analytical thought has gone into implementing such ideas, but we propose additional developments based on our current data mining work and extension of its methodologies (7).

Thus, for our purposes, kill rate can be computed in terms of the probability of hit times the number of shots per individual per unit time. For example, b is the rate at which a Blue combat element destroys a Red combat element, considered derived as a product of the rates of trials and probabilities of kill. That is, in this kind of abstract modeling it is best we consider the “effectiveness” of a unit’s action against other units, as opposed to individual weapon vs. individual target as in our Killer/Victim Scoreboard analyses (8). The unit of measurement for kill rate may be thought of in terms of (casualties of “target force”)/(strength of “firing force” times time). The rate at which fighting occurs may to a considerable degree be measured in terms of the trials with which probabilities of hit and kill are associated. At time t , $f_B(t)$ denotes the average rate at which trials are generated by a single Blue force-unit; $f_R(t)$ is the corresponding rate for a Red unit. For an individual trial, kill probabilities are denoted by k_B for Blue and k_R for Red. The unit killing rates are $c_B = k_B f_B$, the number of Reds that a Blue unit can kill per unit time and $c_R = k_R f_R$, for the number of Blues that a Red unit can kill per unit of time. Attention is confined mainly to homogeneous forces comprising for each side units identical in kill probability and in fighting-rate (9). Moreover, we tacitly consider an integrated “effective” kill rate if the contributions of effects such as reinforcements or disease are not known.

A “plan” or “course of action” for our purposes is defined as: the initial Blue force allocation (a positive number), a Blue breakpoint, a desired kill rate (by Blue of Red), a desired time to win, assumed initial Red allocation (a positive number), an assumed kill rate (by Red of Blue), and an assumed Red breakpoint. We see that there are three types of parameter: known, desired, and assumed. “Actuality” comprises true force allocations, kill rates, and breakpoints. Note that in a more sophisticated formulation, several parametric values can change as part of the plan: e.g., anticipated Blue reinforcement or weakened Red kill rate.

The phase space of the battle is a fundamental notion. Phase space, for the purposes of general discussion, is the n -dimensional space accessible to an object with n degrees of freedom. A somewhat tighter notion important to subsequent technical discussions is that of manifold, a

space locally like n -dimensional Euclidean space, but lacking a preferred system of coordinates and possibly having unusual global topological properties. In most of this decision aid development, we are concerned with the trajectory, the path of an object through phase space. In particular, we are concerned with characterizing the relative positions and movements of the Blue and Red forces in the following technical sense: each force may be thought of as a tuple of coordinates, which may include force level, geographical position, kill rate, ability to move, ability to communicate, etc. For the initial formulation leading to a prototype decision aid, we consider that the trajectory of the battle occurs in a two-dimensional space: the x-axis is the force level of Blue; the y-axis the force level of Red. As indicated elsewhere, this simplistic start results in a number of technical/computational and cognitive/display challenges.

2.2 Design

The technique we are considering would permit comparison of a plan, in terms of desired end state and other measures, with the trajectory of the conflict in phase space. During planning, and particularly during monitoring, if analysis shows the trajectory does not permit the end state, the user would be helped with adjustments enabling the end state or be alerted to the best results achievable. The algorithm involves the following sequence. Input the model form (based on postulated type of operation), parameter values (known, such as Blue force level, and assumed, such as Red break point), and desired end state (e.g., by a certain time cause a certain Red casualty level before Blue suffers a certain level). Compute the resulting trajectory; portray graphically. If the end state cannot be met, assess what changes to parameter values must be made. If a parameter is not established, compute what values will enable the end state. As the (simulated, for this project) operation is monitored, when a value change occurs, recompute the trajectory. Determine if the desired end is still achievable. If it is, reassure the user (perhaps with visual indications of changes); if not, compute the parametric region allowing achievement and enable user trade-offs. Further, during battle monitoring, when new data are obtained, the model itself can be improved: either explicitly, such as by utilizing direct observations of Blue casualties; or implicitly, as in calculating Red attrition rates from force level snapshots. The user is guided to parametric regions allowing end state achievement so that trade-offs can be made if necessary. Values are to be noted as actual or postulated in portrayal of current/projected battle. Initially, for concept demonstration, we utilized a simple model with the intent to then increase the sophistication, eventually monitoring a high-fidelity combat simulation to better demonstrate dynamic model updating.

This is based in part on the idea of iterative wargaming. In this paradigm of battle planning, the analyst sets forth a COA for a relatively short period of time (or until some significant event occurs) and simulates the conflict during that span. Of course, a desired outcome for the conflict and the commander's intent are guiding the choice. At the end of the span, the simulation is stopped, the results tallied, the situation assessed; and the process begins again with a new COA from that point. So, the notion of trajectory correction is similar in that, when a span is reached

in the sense of an updated value for a battle parameter, the situation is assessed for COA changes. The intent of both processes is to bring the long-term goal to fruition if possible.

If a trajectory does not permit the end state, it should be corrected. It is possible that in some formulations and with some ranges of parametric values the end state is mathematically unattainable. For example, the Blue breakpoint might always occur at a time prior to the Red breakpoint with the given model and input ranges. The user should, of course, be alerted to this fact, and the battle would as a practical matter not be entered into.

In the portion of the algorithm concerned with battle monitoring, we shift the intent slightly. Although the concept of checking for ability to reach the desired end state still applies, the nature of the analysis changes to comparison of trajectories. Whenever the commander's staff obtains "intelligence," the system should use this information, if possible, to update the model. In our initial prototype, the updates take the form of snapshots concerning one or the other force level or kill rate. The data can be explicitly ascertained, as in a Blue casualty report that updates the known force level. At this point, the trajectory of the conflict is recomputed, and hence ability to reach the desired end state. If the end state cannot be reached, the necessary changes to parametric values are developed, similarly to the technique in the planning phase.

The new data may also lead implicitly to model updates. An example of this is a new Red kill rate, which could be based on an assumed Blue kill rate, assumed Red force level, and known Blue force level. Techniques for handling such implicit updating have been the focus of a fair amount of research in this project, and their development is discussed in section 7.

Recomputation of trajectory and end state parametric analyses are essentially the same as for the explicit case, at least in our initial model formulation. There is a possibility that, with some model formulations, the implicit case may be handled more efficiently through other methods.

In any event, there is then a "loop" transition back to the earlier monitoring mode. Any update results in a reassessment of the situation as just noted. As discussed elsewhere, updates ideally are handled as they arise, essentially randomly with regard to type, confidence, and time distribution. We have considered in earlier work (10) methods for transforming, basically through interpolation, these into a uniform distribution for ease of computation. Whether this is the best approach from a tactical standpoint, particularly with regard to concerns of levels of indirection, remains to be seen. Notions of sensitivity and computational efficiency also enter into the development. That is, not all updates are of the same utility. Some, for instance, may in effect corroborate the existing model and some may not warrant much additional calculation. The rub, of course, is how to ascertain this in a reasonable manner, given the dynamics of actual combat.

2.3 Decision Aid

The system should develop and plot "envelopes" concerning the trajectory. We had many philosophical discussions among ourselves and with Battle Command Battle Lab personnel

about what this really means. From a technical standpoint, it is not too difficult to calculate “parallel” curves (analogous in some ways to confidence intervals) that bound the trajectory by placing a range of one or more parameters into the mix. It is debatable as to whether this has tactical utility; for example, in enabling the commander to more easily perform risk or sensitivity analyses concerning the feasibility of his plan. Ultimately, we intend this to be the subject of Advanced Decision Architectures Collaborative Technology Alliance (ADA CTA) experimentation.

At any point, we can compare, at least conceptually, planned (desired) Blue and Red trajectories with actual values (or estimated/projected values). Further, we can modify any of these trajectories (separately, or more realistically, together) by changing certain defining parameters, and perform additional comparisons for sensitivity analysis. We would like the Blue commander to receive warnings like “unless the Blue kill rate is doubled by time τ , the goal will not be met.”

The decision aid should also, in a real sense, answer questions the commander might have concerning the planned or unfolding battle. Again, this has been (and continues to be) the subject of much debate. As examples of the kind of “what-if” questions that appear universally desired, we set forth the following. If the enemy maintains this intensity, how much reinforcement do I need at time τ to reach my goal? How much do I need to increase my kill rate or decrease his in order to win? If I desire the battle to be over by some time interval, and know initial troop strengths, what relationship between attrition rates must hold? Such things seem intuitive, indeed even straightforward, yet couching them in forms that are mathematically tractable while militarily reasonable is proving to be a mixture of both art and science.

3. Initial Formulation

3.1 Motivation

As mentioned in subsection 1.3, analysts have used DEs for years to model aspects of combat, in particular attrition. Of particular interest is the possibility of parameterized studies to “optimize” tactics, force deployments, and weapon characteristics. For instance, suppose friendly forces face an enemy in two echelons. Is Red better served by committing the second echelon at some particular time or by continuous reinforcement? How should Blue counter such tactics?

New types of DE models may permit other kinds of investigations. One aspect that might be pursued is application of (multi-variate) control theory to determine (even existence of) optimal responses in combat situations. Perhaps we can characterize through proper variables the state of the battlefield and what might be meant technically by an objective state. Then we can analyze whether the conflict system can be controlled (perhaps via feedback loops) and whether any state

is optimal. Ultimately, such work might be applied to multistage optimization for monitoring and managing battle execution.

This research could also yield reduced complexity for large simulations; sensitivity analyses may show the relative importance of parts of the model, even allowing for reduction of scope if certain items are found to be superfluous. In any event, it would be useful to investigate situations and parameters for which greater “fidelity” may not necessarily be better for the analysis.

3.2 Model

These observations having been offered as motivation, we now consider the initial model, a set of ordinary DEs:

$$\left\{ \frac{dx}{dt} = -ay, \frac{dy}{dt} = -bx \right\}. \quad (1)$$

For obvious reasons, we set forth the constraints $x(t), y(t) \geq 0$. Subsection 4.1 shows why this model is known as the Lanchester “square law” formulation.

The x initial force level and breakpoint are known, the y initial force level and breakpoint assumed, and the time to end desired. The “White” commander (that is, ground truth) inputs initial Red level and the kill rates. The friendly commander inputs initial Blue force level, desired rate at which Blue kills Red, and desired goal state. Examples of goal forms are

$$\{\exists \tau \mid y(\tau) = y_f \cap \tau < \tilde{t}_\omega\}, \quad (2)$$

where \tilde{t}_ω can be infinite and

$$\{\exists \tau \mid y(\tau) = y_f \cap x(\tau) > x_f\}. \quad (3)$$

The Blue commander faces the situation $\{x_0, \tilde{y}_0, \tilde{a}_0, \tilde{b}_0\}$. Given the starting levels, the kill rates, and the breakpoints, strictly determined are the winner (if not a draw, which is unlikely), ending force levels, and battle duration. Our particular desired end state $\tilde{\Omega}$ was defined to be “Red reaches his breakpoint before \tilde{t}_ω (condition 1); Blue reaches his breakpoint after \tilde{t}_ω (condition 2).” In general, we plan based on the initial set $\{\hat{x}_0, \hat{y}_0, \hat{a}_0, \hat{b}_0, \hat{p}_x, \hat{p}_y, \tilde{t}_\omega\}$, where the first four elements drive the trajectory. However, at any time we can change any such parameter and re-evaluate. We now “start the clock,” and the system traces $x(t)$ and $y(t)$. The system also calculates and displays in manners to be discussed later the required parameters when changes in force levels are observed or postulated.

3.3 Sketch of Algorithm

We now build on the discussions of section 2. The prototype development is based on the following sketch of an algorithm, which is broken into four subaspects: inputs, end state determination, correction guidance, and battle monitoring. (Our main development is couched as

the Lanchester square model, but, again, keep in mind there is nothing magic about that particular formulation. It just enables a convenient prototype and facilitates discussion of subsequent concepts.) The reader will see that development of even an apparently straightforward decision aid of this sort offers a number of significant challenges.

In general, we will have a model formulated in terms of a set of equations. Now, in augmenting this set of equations to form a system, we can input “knowns” (e.g., Blue force level), assumptions (e.g., Red force level), and desiderata (e.g., time to end). Input the model, meaning the formulation, the representation of the battle, be it a system of differential equations, a complicated simulation, or whatever. Input the parameter values, meaning the numbers or reasoning symbols that drive the current manifestation of the model, which as indicated, has several components (actual, postulated, desired, modeled). Input the desired end state, meaning the situation in the conflict to which the Blue commander strives and is requesting assistance from the decision aid. We must determine that all parameters are established as constants (although see section 5 on notions such as bounding and what-iffing); initially, this is done through interaction with the user.

Compute the trajectories (meaning the trace of the conflict in parameter space) based on the solution equations, through numerical integration, or whatever method is appropriate. Plot them, or at least the one selected as being of interest in the chosen display (for visualization). Values are noted as actual or postulated in portrayals of current/projected battle. The computation immediately establishes whether the end state can be met. If the end state is achievable, reassure the user; if not, compute the region (which may be complicated, as will be discussed later) permitting achievability in the overall parameter space. The aid must guide the user to determine what changes might be made to bring about the end state. If the end state is mathematically impossible, the user must be so informed; if the aid determines that it is reasonably impossible to complete a required change, then the user should be given a threshold at which to break off consideration.

The next portion, a looping routine, deals with battle monitoring: when a value changes or an unknown becomes a known value, update (improve) the model and recompute the trajectories. The information may be implicit, such as Red kill rate based on assumed b , assumed y , and known x . Again, ability to reach the end state is recomputed. If the end state cannot be reached, the system recomputes the changes needed. The looping for new information then continues.

Let us now discuss in a little more detail certain aspects of the algorithm. The basic initialization subalgorithm first merely asks the user to input what he knows, what he assumes, and what he desires. It then checks whether these inputs are consistent, that is, whether the desiderata are achievable with the given inputs. The user initiates the decision aid program, which identifies itself to the user and cites the time. The program then prompts the user for inputs in the planning stage. First, the knowns: initial Blue force level x_0 , Blue breakpoint p_x (defined as the fraction of x_0 at which Blue stops the conflict), and time to start t_0 (which is used for

planning/replanning purposes in the eventual decision aid; for most of our mathematical development at this point, we can take t_0 as 0 without loss of generality). Second, the assumptions: Red force level \tilde{y}_0 , rate at which Red kills Blue \tilde{a} , rate at which Blue kills Red \tilde{b} , and Red breakpoint \tilde{p}_y (defined as the fraction of \tilde{y}_0 at which Red stops the conflict). Third, the desiderata: in this formulation, time to end \tilde{t}_ω . There are of course many “mechanical” aspects to this. For example, “sanity checking” of breakpoints must ensure these are percentages (and, perhaps, reasonable percentages). We assume that such aspects are handled by the program; ideally, they are foolproof graphic inputs. We also assume for the prototype that all these basic inputs are established; however, dealing with uncertainty in the values must form an important part of an actual system.

The program computes Blue and Red trajectories, based on the strictly-determined solution equations, and plots them starting at t_0 and ending at the highest time of interest. Figure 2 illustrates this schematically. The plot shows time as the abscissa and force level as the ordinate. Moreover, the breakpoints are cited as both time and level for both x and y forces; \tilde{t}_ω is given as a vertical line.

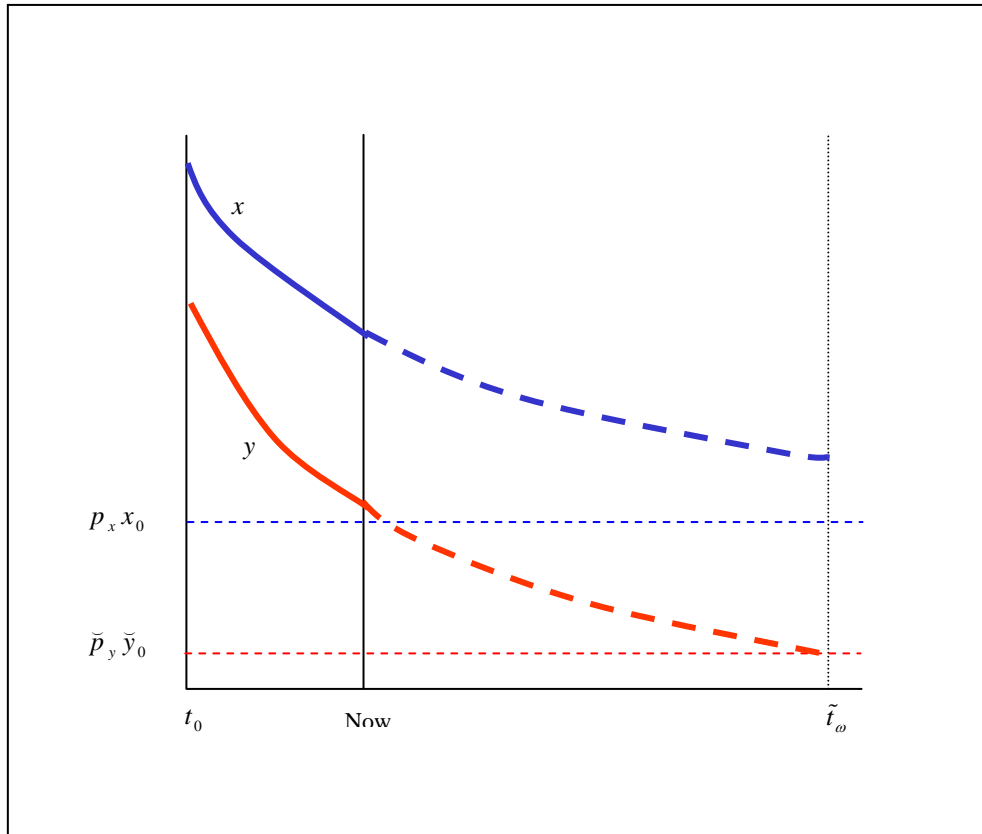


Figure 2. Blue and Red force levels over time.

Let us call the modeled time at which Blue reaches his breakpoint \hat{t}_x : $x(\hat{t}_x) = x_0 p_x$, where we have taken advantage of the monotonicity of the trajectory. Similarly, we have $\hat{y}(\hat{t}_y) = \tilde{y}_0 \tilde{p}_y$.

The program checks whether $\tilde{\Omega}$ is achievable: condition 1 of $\hat{t}_y < \tilde{t}_\omega$ and condition 2 of $\tilde{t}_\omega < \hat{t}_x$. If these conditions are met, the program so informs the user.

Now the plot thickens, so to speak. If condition 1 is not met, we want the decision aid to guide the user to correct the situation for planning. More-or-less obvious suggestions can be made involving the “raw variables” end time and Red breakpoint; for instance, relax the end time to accommodate the calculated value. (Such suggestions actually turn out to be somewhat less obvious than what might be thought, due to their dependency of these on the larger “mix” of variables. This will be considered later in more depth.) However, we want more sophistication for our prototype decision aid.

In order for trajectories to have value in what-iffing and sensitivity analyses, we must have a way to compare them. Comparisons based on time would seem relatively straightforward, and indeed, as a practical matter, this is a tactically reasonable way to proceed. That is, are the desired parameter values plotted over time as the commander desires? (This will be examined in terms of force levels in the initial formulation. However, deviation from plan is difficult to assess even in this simple case, as will be seen.) We might consider comparing parameters without the notion of time. This at first seems more abstract as far as tactical applicability. However, meaningful comparisons can be made; again, as will be seen in the initial formulation when we examine relative force levels as determining the outcome.

One question the friendly commander will ask is, “What can I do to move the enemy breakpoint time to where I want it?” Since he has no particular influence on y_0 or \tilde{a} (other than by changing his assumptions, which we consider in a subsequent discussion of sensitivity), we see we may want to solve for x_0 or \tilde{b} in terms of desired enemy breakpoint time. At this point, we need to consider some mathematics.

4. Theoretical Preliminaries

4.1 Fundamental Notions

We begin with some terminology. Let us consider a system

$$\{\dot{x} = f(t, x, y), \dot{y} = g(t, x, y)\}, \quad (4)$$

with $x(0) = x_0$ and $y(0) = y_0$, where both f and g have continuous first partial derivatives. The functions $t, x = x(t)$, and $y = y(t)$ in an interval define a solution if

$$\dot{x}(t) = f(t, x(t), y(t)) \quad (5)$$

and

$$\dot{y}(t) = g(t, x(t), y(t)) \quad (6)$$

for all t in the interval. Now the (x, y) -plane is called the phase plane and the parametric curve $\{x = x(t), y = y(t)\}$ is the trajectory of the system. If the points of a trajectory are in one-to-one correspondence with the parameter t , the direction in which t increases is called the positive direction; arrows are sometimes used in plotting to indicate this. A collection of trajectories is a portrait.

Generalizing to higher dimensions, the vector $\mathbf{x}(t)$ traces the trajectory in the n -dimensional phase space of the state vectors, where $\mathbf{x} = \mathbf{x}(t)$ is a solution of $\dot{\mathbf{x}} = f(t, \mathbf{x})$ in an interval. The point $(t, \mathbf{x}(t))$ traces a time-state curve in the $n+1$ -dimensional space of the time and state-variables. The projection of this curve onto the (t, x_i) -plane is the x_i -component graph. Note that in subsequent discussions we may speak of component graphs or other functional traces of parameter values as “trajectories,” the meaning being clear in context.

An autonomous system has the property that only initial position and elapsed time are important in phase space: if

$$\{ \mathbf{x} = \mathbf{x}(t), \alpha < t < \beta \} \quad (7)$$

is a solution, then so is

$$\{ \mathbf{x} = \mathbf{x}(t+k), \alpha-k < t < \beta-k, k \text{ constant} \}, \quad (8)$$

and the trajectories are seen to be identical. Whether combat in general is an autonomous system is open to debate; but note that t does not explicitly appear in equation 1, so our initial model is autonomous. Anyway, it is apparent autonomous systems are simpler to analyze.

Continuing to look at our model, since dx/dt is not equal to zero in the first quadrant (the region of tactical interest), we find

$$\frac{dy/dt}{dx/dt} = \frac{-bx}{-ay} \quad (9)$$

or

$$ay \frac{dy}{dx} = bx, \quad (10)$$

where time has, in effect, been replaced by x as the independent variable. Put another way, the initial formulation yields the ratio of the two equations in the system: $\frac{dx}{dy} = \frac{ay}{bx}$. Now writing this as

$$\int_0^{\tau} bxdx = \int_0^{\tau} aydy \quad (11)$$

yields

$$b(x_0^2 - x^2(\tau)) = a(y_0^2 - y^2(\tau)), \quad (12)$$

or

$$ay^2(\tau) - bx^2(\tau) = ay_0^2 - bx_0^2, \quad (13)$$

hence the characterization of equation 1 as the “square law” model (11).

4.2 Analytical Results

The so-called “state equation”

$$b(x_0^2 - x^2(t)) = a(y_0^2 - y^2(t)) \quad (14)$$

(which does not explicitly consider time as a variable) is in itself useful for deriving results of potential utility in our analyses. For instance, consider the force level ratio $r = x/y$. It turns out this ratio satisfies

$$\frac{dr}{dt} = br^2 - a. \quad (15)$$

This equation provides qualitative information about the force ratio: x is winning if and only if $x/y > \sqrt{a/b}$; a necessary and sufficient condition that x wins a battle to annihilation in finite time is

$$\sqrt{a/b} < x_0/y_0 \quad (12).$$

That is, battle outcome is determined by the initial force ratio and the relative attrition, even though there are four parameters in the model.

Again, writing

$$ay^2(t) - bx^2(t) = ay_0^2 - bx_0^2, \quad (17)$$

and calling the right side constant c , note that if $c < 0$, then x wins, since x does not vanish, but $y = 0$ when $x = \sqrt{c/a}$. Blue wants conditions in which $\left(\frac{y_0}{x_0}\right)^2 < \frac{a}{b}$. Due to the quadratic nature of this inequality, note that a small increase ε in y_0 could change the predicted win to a loss:

$$\left(\frac{y_0 + \varepsilon}{x_0}\right)^2 > \frac{a}{b} \quad (13).$$

Such calculations form the basis for the dynamic decision aid. Note also the notion of “basin” (in the sense of qualitatively different areas of the space) that surfaces in this last observation concerning ε .

Similarly, the state equation can be manipulated to derive solutions for the force levels as functions of time:

$$x(t) = x_0 \cosh \sqrt{abt} - y_0 \sqrt{\frac{a}{b}} \sinh \sqrt{abt} \quad (19)$$

and

$$y(t) = y_0 \cosh \sqrt{abt} - x_0 \sqrt{\frac{b}{a}} \sinh \sqrt{abt}. \quad (20)$$

Further, we have

$$\frac{x(t)}{x_0} = \cosh \sqrt{abt} - \frac{y_0}{x_0} \sqrt{\frac{a}{b}} \sinh \sqrt{abt}, \quad (21)$$

showing the force level depends on three derivative parameters: ab , y_0/x_0 , and a/b . It is seen that ab controls the length of the conflict. Such derivations have been pursued in great detail by several researchers, in particular, by Prof. Taylor (12). However, for decision aiding we must consider more realistic conditions such as breakpoints and desired time to finish.

Consider the four main inputs to the model: x , y , a , and b . These all impact the modeled Red breakpoint time, which we shall denote \hat{t}_y . Let us solve for \hat{t}_y , and then consider the change in each variable, at first separately, that will result in condition 1 being met. Later, as an improvement to the decision aid, we will explore the tradeoff in parameter space that will enable this. Since $\cosh x = (e^x + e^{-x})/2$ and $\sinh x = (e^x - e^{-x})/2$, some algebra and the reasonable assumption $\hat{t}_y \gg 0$ yield

$$\hat{t}_y \approx \ln \frac{2\tilde{y}_0\tilde{p}_y}{\tilde{y}_0 - x_0\sqrt{\frac{b}{a}}}. \quad (22)$$

We now have a relation between the modeled enemy breakpoint time and the four Lanchester parameters. We can now develop sliders involving multidimensional plots of the parameter space under these conditions. For example, we can fix $\hat{t}_y = \tilde{t}_\omega$, y_0 , and \tilde{a} ; and then show a plot of x_0 vs. \tilde{b} as an indicator of the tradeoff between initial force level and kill rate required to bring about the end of the conflict by the desired time. (This, of course, assumes condition 2 is met. We must at some point in the tradeoff analysis check whether the Blue force is diminished to too low a level at the desired end time, clearly an unacceptable situation.) Alternatively, we find

$$\tilde{b} \approx \tilde{a} \left[\frac{\tilde{y}_0}{x_0} e^\tau (e^\tau - 2\tilde{p}_y) \right]^2, \quad (23)$$

where τ here indicates the desired time of enemy break, computed based on the model solution

$$y(\tau) = p_y y_0 = y_0 \cosh \sqrt{ab} \tau - x_0 \sqrt{\frac{b}{a}} \sinh \sqrt{ab} \tau \quad (24)$$

and again assumed $\gg 0$.

Let

$$m = y_0 / x_0, a = k^2 b, \quad (25)$$

with all quantities positive. Based on equation 19, the Blue strength level at any time can be written as

$$x(t) = \frac{1}{2} \left[(x_0 - \sqrt{\frac{a}{b}} y_0) e^{\sqrt{ab} t} + (x_0 + \sqrt{\frac{a}{b}} y_0) e^{-\sqrt{ab} t} \right], \quad (26)$$

which implies

$$x(t) = \frac{x_0}{2} \left[\left(1 - \sqrt{\frac{a}{b}} m\right) e^{kbt} + \left(1 + \sqrt{\frac{a}{b}} m\right) e^{-kbt} \right]. \quad (27)$$

Note that as $t \rightarrow \infty, e^{kbt} \rightarrow \infty, e^{-kbt} \rightarrow 0$. Moreover, $1 \pm \sqrt{\frac{a}{b}} m$ are constants. If x loses (i.e., $x \rightarrow 0$), then it is necessary that

$$1 - \sqrt{\frac{a}{b}} m \leq 0 \Rightarrow \frac{a}{b} m^2 \geq 1 \Rightarrow a \geq \frac{b}{m^2}. \quad (28)$$

Similarly, if $a < \frac{b}{m^2}$, then Blue wins (i.e., $y \rightarrow 0$). Also, note the further one deviates from this critical point where $a = \frac{b}{m^2}$, the more decisive the win. Critical values for a and b such that $a = \frac{b}{m^2}$ result in a battle where both forces essentially go to 0. Values of a greater than $\frac{b}{m^2}$ result in Blue losing, and of a less than $\frac{b}{m^2}$, Blue winning.

Let $km > 1$, i.e., Blue loses. Rewriting equation 27 using k, m , and x_0 , and letting $E = e^{kbt}$, we get

$$x(t) = \frac{x_0}{2} \left[(1 - km) E + \frac{(1 + km)}{E} \right]. \quad (29)$$

Blue may decide to end the battle when its strength reaches a level less than x_0 , say $p_x x_0$, where $p_x \in [0, 1]$. Then we have the equation

$$p_x x_0 = \frac{x_0}{2} [(1 - km)E + \frac{(1 + km)}{E}]. \quad (30)$$

If we multiply equation 30 by $-E$ we get

$$(km - 1)E^2 + 2p_x E - (1 + km) = 0. \quad (31)$$

The roots of this equation are

$$\frac{-p_x \pm \sqrt{p_x^2 + k^2 m^2 - 1}}{km - 1}. \quad (32)$$

Since $E = e^{kbt} > 0$ and the root $\frac{-p_x - \sqrt{p_x^2 + k^2 m^2 - 1}}{km - 1}$ is less than 0 for $km > 1$, it cannot be a valid root. Thus, the only root of interest is

$$E_0 = \frac{-p_x + \sqrt{p_x^2 + k^2 m^2 - 1}}{km - 1}. \quad (33)$$

In the event x loses, values for $p_x \in [0, 1]$ will give a real root of equation 33. In the event x wins, then the x strength level does not fall below some critical value $p_{xc} x_0$, while the y strength falls to 0, and a root exists if $p_x \geq p_{xc}$; otherwise, no root exists. A similar argument holds for y . We can then solve the equation $E = E_0$ or

$$\exp(kbt_\omega) = E_0 \quad (34)$$

to obtain

$$\tilde{t}_\omega = \frac{\log_e E_0}{kb}. \quad (35)$$

In general, equation 30 can be solved numerically for one of the variables $\{p_x, k, b, t\}$, given the other three. The decision aid might use such calculations to help the commander address such situations as the following. I am involved in a battle, but by t_e time intervals I need to be elsewhere to support another battle. Having determined I can inflict casualties on the enemy at rate b , what casualties can I accept in order to win the battle within t_e time intervals and retain at least $p_e x_0$ of my forces? The question can be answered by using equation 31 to find the attrition rate a corresponding to the strength needed and equation 34 to determine whether the mission can be accomplished within t_e time intervals.

We see that the algorithm can be modified to give the maximum attrition rate that allows mission accomplishment: answering this question reduces to solving equation 31 for k , where $a = k^2 b$, yielding the maximum attrition that allows mission accomplishment. A related situation is: if Red is winning the battle, at some point in t_e time intervals into the battle what could happen in

terms of force strength or attrition rates so that Blue wins? The critical inequality to consider is: Blue wins if $a < b/m^2$, i.e., if $ay^2 < bx^2$. A Blue win occurs by an increase in strength of

$$\Delta x > \sqrt{a/b} y_{t_e} - x_{t_e}. \quad (36)$$

Alternatively, if Blue can decrease Red's attrition rate by

$$\Delta b > \frac{ay_{t_e}^2}{x_{t_e}^2} - b, \quad (37)$$

or if Blue can increase its own rate by

$$\Delta a > a - \frac{bx_{t_e}^2}{y_{t_e}^2}, \quad (38)$$

then Blue will win.

5. Aspects of the Decision Aid

5.1 Introduction

We now examine this work from a somewhat different perspective. While still technical, this section is less concerned with mathematical modeling and computational algorithms. It sets forth a variety of human-centered display considerations for research and development in pursuit of an actual prototype decision aid for model-based execution monitoring. We postulated various displays that show the past, present, future; real, assumed, desired, modeled factors; boundaries, confidence levels; and other information of interest to the commander. Almost any one of these could be a project in itself, and we intend that they be approached in an orderly manner by a design/programming team.

At some point in monitoring the battle, we may receive new information. As sketched earlier, we can input the updated parameters and compute a new trajectory from that point (and check that the desired outcome is obtainable). We use the former model's values at that time to form the non-new information, such as Red strength level. A more complicated computation would involve fitting the form of the model to the observed data, and this sort of approach is discussed in some detail in section 7. A more sophisticated interface would be required to explore the effects of model formulation on conflict trajectory. Curve fitting or regression software might be leveraged for both these purposes. An abundance of information is available on curve fitting, and many of the software packages mentioned in section 6 utilize curve-fitting algorithms.

Continuing in a bit more detail, let us again distinguish among model values; that is, consider separately the "best estimates" (e.g., $\hat{x}(t)$) and the actual reported values (e.g., $x(t)$). Let us now

consider the battle to be on; the system is in execution monitoring mode. At time τ , the Blue commander receives a report that $x(\tau) = x_\tau$. If $x(\tau) = x_\tau = \hat{x}(\tau)$, he is on track with regard to the plan playing out properly. If $x(\tau) > \hat{x}(\tau)$ (and all other things being equal – which they are probably not, as discussed elsewhere), he is doing better than expected with regard to this parameter of interest. The system now modifies the working model (or more properly for the initial prototype, its set of parameter values) to reflect this latest strength information in the solution set

$$\begin{cases} \hat{x}(\tau + t) = x_\tau \cosh \sqrt{ab}(\tau + t) - \hat{y}_\tau \sqrt{\frac{a}{b}} \sinh \sqrt{ab}(\tau + t) \\ \hat{y}(\tau + t) = \hat{y}_\tau \cosh \sqrt{ab}(\tau + t) - x_\tau \sqrt{\frac{b}{a}} \sinh \sqrt{ab}(\tau + t) \end{cases} \quad (39)$$

A new estimate of the end state is also calculated, and if the Blue commander is not satisfied with the situation, he can estimate the consequences of changes in parameter values.

Similarly, if $x(\tau) < \hat{x}(\tau)$, the commander is in trouble (all other things being equal). Again, the model is modified and a new estimate of the end state calculated (presumably Red will win or Blue will be too late). What can the commander do to “correct” the situation? We might assume $x(\tau)$ must be increased to $\hat{x}(\tau)$ in order to get back on track; however, a problem arises due to the probability that Red is now better off, so computations involving $y(\tau)$ as well now become germane.

At any time, we should be able to display graphically for the commander various aspects of the trajectory spaces. Of fundamental importance is what region he is in, in order to assess whether he is meeting his goal with respect to his chosen criterion or criteria. Of course, auxiliary windows should indicate position in different parameter spaces. The commander would usually access such displays purposefully, but if some dangerous situation arises that requires alert (perhaps as determined by methods discussed in section 8), the appropriate auxiliary alert window should be displayed automatically.

The commander can also be presented with options given a fixed parameter. That is, the commander would choose a value for “parameter 1” and be presented with a curve in the space defined by “parameter 2 (abscissa) and parameter 3 (ordinate)” yielding that value. Multiple such curves should be plottable on the same set of axes.

5.2 Trajectory Tracking

It is fundamentally important to be able to plot trajectories under various assumptions. These could generally be displayed in sets, perhaps analogously to iso-surfaces. The best way for the user to immediately grasp information being portrayed is an area of active research that is beyond the scope of this effort. However, the reader is referred to pioneering work being done at the Institute for Human and Machine Cognition (14).

A related aspect is computation and display of rates of divergence of the actual trajectory from that planned. For simple trajectories and concern with a single parameter, such divergence can be handled in a basically intuitive manner. However, determining what this really means in a stringent technical sense turns out to be somewhat difficult. It is hoped that subsequent research, probably utilizing Lyapunov analysis (discussed in subsection 8.4) will shed more light on this problem.

Another aspect of the trajectory of concern to the planner and execution monitor is the rate of change along the chosen trajectory. Auxiliary concerns arise analogously to those just discussed. An area of research addressable by cognitive scientists in the ADA CTA (15) is whether transformation to the first derivative should be done by the computer prior to displaying such information. That is, are points in a “derivative state space” easier for the user to interpret than following the actual motion along the trajectory? In any event, information such as time to reach the boundary of a critical region if no parametric values are changed must be developed and displayed to the user.

An even further extension is consideration of “acceleration” in phase space. That is, analysis of second derivatives could help the commander with questions of criticality concerning model parameters, particularly with regard to time. For instance, if the approach to an undesirable region is increasingly rapid, determination of parameters and correction of values becomes of paramount importance. One issue that arises in such extensions is that of meaningfulness. That is, as the analysis of the conflict takes on more degrees of indirection, it perhaps becomes less relatable to the actual tactical situation. At the very least, the notion of sensitivity vis-à-vis numerical analysis is germane if only in a technical sense.

One relatively easily developed plot is that of Blue (Red) trajectories as a function of Red (Blue) initial force level. With an appropriate graphical user interface (GUI), the user can explore times to reach certain levels as either “dependent” or “independent” variables. This technique also illustrates the notion of a simple possibility envelope, achieved by setting bounds on the assumed initial level.

Another approach to the overall system design shows some promise: develop an allowable envelope and track the actual trajectory in that context. Figure 3 illustrates this schematically; the actual trajectory is plotted, with alerts shown as the envelope is intersected. This seems particularly attractive in light of the desire expressed by the U.S. Army Battle Command Battle Laboratory that the user be provided not only an alert but also what to do about it. Of course, this could be made more sophisticated by incorporating simple devices such as having the past plot fade with age or by color-coding the breaks according to exit/enter top/bottom. Further, the user could be warned and guided based on derivative analysis of apparent approach to the envelope. Time to reach the boundary of a critical region if no parametric values are changed could also be displayed.

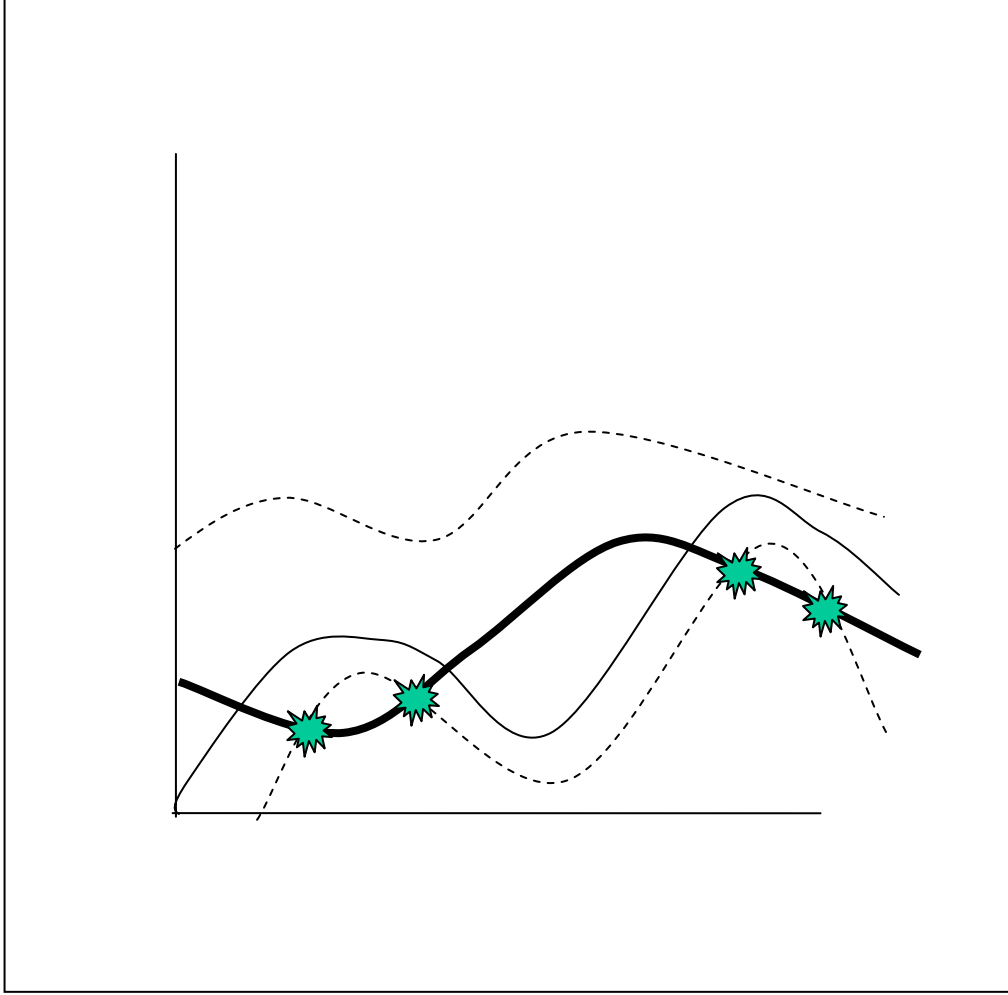


Figure 3. Planned trajectory with alerts as actual trajectory breaks envelope.

As mentioned earlier, we can plot various aspects of the battle. In particular, in our Lanchester formulation, we can look at the force ratio trajectory. This trajectory is of particular interest, since it is readily indicative of battle-state regions, and is a good starting point for further work on nonlinear dynamics in the next formulation.

5.3 Sensitivity Analyses

We now have, based on the given initial parameters, plots of the Red and Blue trajectories and an assessment of whether the enemy can be defeated. Another form of decision aiding can be sensitivity analyses in the form of overlaying on the “real” trajectories variations in which certain parameters are “dithered.” For example, the commander should be able to have sliders in which he changes x_0 , \tilde{b} , y_0 , or \tilde{a} singularly or in multiple fashion. (Such notions will be discussed in more detail shortly.) The manifestation on the plot would be shifting trajectories in which the user can see the influence. Of course, the breakpoints can also be changed, resulting in moving of the “crosshairs.” All these are readily computable from the solution equations.

Various kinds of sensitivity analyses appear to be useful to the staff. Some of these are discussed in other subsections. One way to perform them is having the user choose a parameter(s) to vary, with the others defaulting to (pre)selected values. With a reasonable GUI, the user can easily and dramatically explore interactions among the model parameters, such as the influence of enemy initial strength on friendly loss. Derivative aspects of the trajectory would seem to be worthwhile, especially in an extremely time-critical environment. This notion includes rates of change of aspects of the battle, and how they influence the outcome. Dithering certain values and rerunning (or running in parallel) the repercussions would seem to be useful, especially for logistical analyses. This notion includes consideration of possibilities of bad intelligence. As an example, the commander is probably immediately interested in analyses of what happens if the enemy force is 10% larger than he is estimating for the game. Again, the actual tactical utility of such capabilities remains to be measured.

A more sophisticated guidance mechanism would indicate to the user the qualitative nature of his proposed changes. For example, given that the user wishes to “raise” the blue curve, there should be a plot in n -dimensional parameter space showing the region(s) in which this occurs. We considered natural language type conversations and symbolic aspects of the system to be beyond the scope of this project. However, we will consider a simplified version of this in order to explore the navigational aspect of the decision aid. Let us have a tactically reasonable input for the user: desired end time, in the sense of Red break point manifestation. The trajectories are strictly determined, and the end time can be slid back and forth over the graphs, with a changing read-out of p_y , where all other parameters are fixed. This changing readout has nothing magic, though, about p_y ; by fixing p_y and selecting, say, x_0 , the sliding end time can show the changing read-out in terms of Blue initial force level required to bring about the desired end time. This can be done for any of the parameters; there is a color-coded indicator to the user of given/assumed in the plots and numerical readouts. It is, of course, up to the user to exercise proper military judgment in assessing the options resulting from such “dithering” of the decision aid.

We plot $x(t)$ and $y(t)$ from t_0 to, say, twice the input p_y . (Again, a nice aspect of the graphical decision aid would be to have this window stretchable and automatically scaled.) The input delimiters (e.g., Blue breakpoint) are plotted with both vertical and horizontal lines dropped to the axes. The user inputs a time to end \tilde{t}_ω by clicking on the plot; a vertical line appears through that time, and horizontals as well from the force level plots. All abscissa/ordinate pairs are displayed on the plot, having been immediately calculated from the equations for $x(t)$ and $y(t)$. An auxiliary screen appears. In it is displayed the p_y associated with the time to end \tilde{t}_ω ; this is computed easily from $y(\tilde{t}_\omega) = p_y y_0$. Now the user can drag this vertical line left or right, and the changing p_y is computed and displayed continuously, with the slider “stopped” when p_y reaches 0 or 100%.

Now, let us thicken the plot, so to speak. What is the display like if there are two variables being changed? Let us take a specific example again. We select p_y and x_0 as the changeable variables.

Now each position of sliding the desired end time results in a locus of pairs (p_y, x_0) that produce the desired result. Since for any selected τ we have the model solution (assuming the original parametric values) as

$$y(\tau) = y_0 \cosh \sqrt{ab}\tau - x_0 \sqrt{\frac{b}{a}} \sinh \sqrt{ab}\tau \quad (40)$$

$$(\text{and } x(\tau) = x_0 \cosh \sqrt{ab}\tau - y_0 \sqrt{\frac{a}{b}} \sinh \sqrt{ab}\tau). \quad (41)$$

Note that these make tactical sense only until one side is annihilated. We can now compute from the y -trajectory the function

$$x_0 = y_0 (\cosh \sqrt{ab}\tau - p_y) \left(\sqrt{\frac{b}{a}} \sinh \sqrt{ab}\tau \right)^{-1}. \quad (42)$$

This curve can be shown in an auxiliary screen; the curve changes as the slider moves, changing τ . Note that there is a relationship between p_y and x_0 , rather than having them strictly determined. This may seem strange because, after all, we have two equations in two unknowns for any given τ . However, the link between the trajectories for x and y is strict, due to the nature of the initial model previously analyzed. Moreover, we had no problem considering p_y in the one-variable case. This gives a kind of reasonable assistance to the commander, in that he can see what his initial force level must be in relation to the Red breakpoint, given a desired end time. Of course, we must still check for a reasonable x trajectory.

Continuing, let us consider three variables as changeable: we select p_y, x_0 , and \tilde{b} . Each position of sliding the desired end time results in a locus of triples (p_y, x_0, \tilde{b}) that produce the desired result. This surface can be shown in an auxiliary screen, with the surface changing as the slider moves. Now we enter the realm of information displays per se. For example, software can be utilized/developed that enables the user to rotate the surface for better examination. Another possibility is to let the user fix one of the three variables and produce a curve display as previously shown. However, again, we considered this beyond the scope of the project.

A desirable feature of a COA in general is that it be “robust.” This is usually interpreted to mean that the plan produces good results even in the face of widely differing enemy COAs, and is assessed by wargaming against a most likely and a most dangerous enemy plan. In this model-based decision aid, we would like to continue this notion of robustness, but probably in the form of risk assessment. It seems that sensitivity analytical techniques can be used to examine subtleties in the battle construction that could result in “squeakers” or the onset of chaos (a notion discussed in subsection 8.3). These techniques can be either closed-form analytical for relatively simple systems like the prototype uses or numerical analytical (or even simulation-based) for more complicated models. Development of what might be called confidence bands is

a worthwhile extension of the notion of trajectory envelopes: a new idea is to use deltas in parameters to yield bounds and hence probabilities.

5.4 Qualitative Behavior

Let us return to the notion of examining the qualitative behavior. As a practical matter, the user may not be looking for exact values of the changeable variables (at least not at first); the user just wants to see whether increasing one of the parameters, say, or increasing one and decreasing another, brings about the desired result. We now enter the realm of partial derivatives. For example, go back to the initial example where we were sliding the end time and looking at the changes of p_y . If we differentiate the solution curve(s) with respect to that variable, we have another equation that can be examined similarly. But this one is in a sense more straightforward, in that we have natural regions of positive and negative (“slope”) behavior separated by zeroes. This can be transformed back automatically for the user onto the original plot showing where the variable produces the desired change.

This idea is generalizable. Suppose the user wants to examine the effect on certain variables within the model of changes on other variables (where the “change” may involve fixing it as a constant). This requires a shift in thinking to considering the model as a relation or set of relations among the variables.

The notion of “possibility envelopes” is a bit harder to describe. Perhaps it is best to think of it as examining a “tradeoff space.” It is basically a kind of inverse problem, where the conditions that would bring about a desired situation must be derived. An example might be to calculate and display a function of several parameters (a curve or surface in three-space, for example) such that any point in this set yields the desired result or range of results, given a selected model formulation.

If any of the enabling conditions is violated, a yellow region can be visualized; if violated beyond some ε , a red region. Of course, it is problematic how to assess ε scientifically, as opposed to just having the user make educated guesses. It is important that the system assess at every time (or “phase” of the battle, in the sense of agreed-upon tactically-significant events) what conditions would result in not meeting the goal. For example, from the perspective of (τ, x_τ) what region of (y, a, b) -space causes: (1) defeat of Blue (a red situation), (2) longer time to win than desired (yellow situation), and (3) more Blue casualties than desired (yellow situation). Such assessments, if properly visualized (perhaps overlaid with probability curves, another challenge), can help the commander assess risk/sensitivity and needs for intelligence or change of tactics. If a slight increase in a would cause many more Blue casualties, we may want to reassess our COA. Visualization may be based on a fixed (τ, x_τ) , for example, so that the desired region is a kind of “floating rectangular solid.” A more sophisticated development, but still based on the mathematics of the model, would enable the user to click on any point in this space to get the projected time to defeat or lose, Blue casualties, and so forth. Another type of

visualization is exemplified by fixing a at a chosen value and having the system plot curves of constant time to end.

So we see how we can visualize red and yellow situations for a given model. How do we compare plan and reality? For a desired end state and a given set of fixed parametric values, we can assess our ability to reach the end state. We can click on a point in red/yellow regions to check on values of the orthogonal space that might enable the desired end. We can play the plan model in real time and note if snap-shotted actual-combat values fall into red/yellow regions. But it would be most valuable to track the motion of these values, warn the user if any is headed into a danger region, and offer suggestions about possible corrections. Moreover, these concepts are intimately related to the techniques for analyzing dynamic systems. The philosophy and mathematics behind such a system is discussed in more detail in section 8.

5.5 Other Desiderata

Obviously, the ability to display information about all the various parameters is essential. This includes human-centered displays of values (whether actual, assumed, or modeled), relationships among them, allowable ranges, and so forth. A related notion, previously alluded to, is that of implementing mouse sensitivity of various objects; moving the cursor over certain regions or clicking points on a given curve/axis would automatically display more information about that object. For example, choosing the projected portion of a trajectory might display the actual and assumed parametric values that yield it. Moreover, many aspects can be animated, with colors indicating the nature of the portrayed information, such as modeled based on real data, assumed data, or both.

At a given time, there are generally a set of conditions that guarantees the Blue goal, a set guaranteeing the Red goal, and a set guaranteeing a “draw.” Note that we say “generally.” Exhaustive characterization/determination of such sets for an arbitrary model is important follow-on work. As mentioned, determining reasonable techniques to display these conditions is ongoing research in several arenas.

It is considered desirable in some circles to show the user the battles. That is, a graphical portrayal of the predetermined simulation and/or the unfolding conflict is displayed on a computer screen filled with tactical overlays. This aspect is somewhat tangential to the intent of this initial work. The model-based results of the prototype decision aid could, however, be tied into such a display by future engineering. It is hoped that some emerging results of the ADA CTA can be leveraged in this regard.

A related notion is to tie in our work on developing and visualizing “terrain ownership.” Again, the idea would be to augment the model-based decision aid results concerning the simulated and actual battles by providing the user with a distilled portrayal of realizable combat power in the area of operations (7).

Many of the sorts of engineering details we have been discussing may seem to be trivial mechanical implementations. Once the mathematical underpinnings are developed for the model-based execution monitoring, however, such display aspects are vital to enable a reasonable decision aid, and should be developed in conjunction with the other analytical aspects. Indeed, we see that many of them are complex mathematical and cognitive challenges in themselves.

6. Software

6.1 Consideration of Options

Based on the developments previously mentioned, there naturally arose several aspects of the project to which we thought to apply commercial software packages: differential equation solvers, curve fitters, and data plotters for experimental mathematics. We performed a fair amount of investigation into the availability of software both for solution of systems of differential equations and for display and evaluation of data in a GUI. We considered the particular design applications arising, and looked in this context at a variety of factors such as ease of use, maturity of development, reputation, and expense.

DE solvers can be based in some senses on algorithms set forth in the Numerical Recipes series (16). Ordinary DE problems can always be reduced to sets of first-order DEs. New variables can be derivatives of each other and the original variable. Sometimes, one can incorporate some power of the independent variable or other factors; different auxiliary variables can often be chosen to mitigate computational difficulties. Solutions can be obtained in finite form for equations of all orders having linear constant coefficients. Infinite series solutions can be obtained for many linear equations with variable coefficients. Good numerical approximations can be obtained for equations whose complexity makes them analytically intractable. It is probably true that all solutions of any (solvable) system can be found with reasonable accuracy. However, as alluded to earlier, many questions concerning a system can be addressed by considering only descriptive properties. One such notion, stability, will play a vital role in future investigations.

We considered many systems, including: MATLAB, Maple 9, LabVIEW 7 Express, OriginPro 7, Gauss 5.0, Scientific WorkPlace, FlexPDE 3, Nmath Core, Interactive Data Language, IRIS Explorer 5.0, PV-WAVE, JMP 5, STATISTICA, Mathcad, Mathematica, S-PLUS, DPGraph, ActivityBase 5.1, JMSL, Diffpack, Tecplot, and EnSight. We even considered certain imaging software, but found it generally too specialized to image analysis per se. (An exposition of the plusses and minuses of these packages might be a useful derivative effort to this project.) As the reader can see, much such software is available as commercial packages. Of

course, many existing Java tools and applets lend themselves to interactive plotting, as may be seen by a few Google searches.

Discussions with various graphics and scientific visualization experts were useful in narrowing down the possibilities. Some of these discussions produced somewhat unexpected results. For instance, MATLAB certainly appeared to be a good possibility. It includes DE solvers, multidimensional data support, and interactive capabilities for 2-D plots and 3-D surfaces. It enables customization of functional source code and linking to external software and data. However, discussions with the U.S. Army Research Laboratory (ARL) scientific visualization group were discouraging with regard to ease of use of the GUI and lack of availability of a current MATLAB “guru.” We similarly eliminated S-PLUS, although useful for canned statistical evaluations. We participated in an ARL seminar on Analysis and Visualization of Large Data Sets that helped us with visualization techniques, and a review of Major Shared Resource Center projects resulted in additional knowledge concerning high-performance computing software that might be leveraged.

One possibility for enabling a visualization environment template for the prototype, and one that is a natural follow-on to collaborative work already being performed in the ADA CTA, is to utilize the Seeker-Filter-Viewer technology of the Ohio State University Laboratory for Artificial Intelligence Research. This system, described extensively in various references (17, 18), utilizes a Java-based simple GUI by which the commander can specify what data to see in what kind of display. Another opportunity for collaboration is via work being done at the Center for the Representation of Multi-Dimensional Information (19). We have been interacting with Prof. Foresti at the University of Utah and, given a proper state of maturity, intend to utilize the sophisticated software developed by his multidisciplinary team.

Various software packages exist for exploring nonlinear dynamics. All require a graphics display (such as video graphics array, enhanced graphics adapter, or color graphics adapter), but that should not be a real problem. Chaos, the Software, by Rucker, Autodesk Inc., illustrates various ideas of chaos such as the Mandelbrot set, attractors, magnetic pendulum, and fractals (20). INSITE is a software package of graphical interactive programs useful for examining nonlinear systems. Written in C and running in DOS or UNIX, it is a possibility for future algorithmic development (21). Phaser, by Kocak, is companion software to one of our references (22). A fairly complex program that covers a range of systems and iterated maps, Phaser permits the user to analyze specific difference/differential equations of theoretical or practical importance with regard to their dynamic behavior. It has been used for undergraduate courses at Brown University, and has sophisticated interactive graphical capabilities that make it a useful exploratory research tool. However, it is somewhat dated. We have found another tool, discussed in the following subsection, that appears well-suited to our work.

Physics Academic Software makes available several dynamics packages. Mapper, by Harold, computes trajectories, Lyapunov exponents, and Poincaré maps. It allows entering one’s own

equations and handles 2-D iterated maps and DE systems. Chaotic Dynamics Workbench, by Rollins, generates state space diagrams, Lyapunov exponents, and Poincaré maps. It handles systems described by a few ordinary DEs and permits changing parameters and examination of transients. Chaos Demonstrations, by Sprott, permits adjustment of certain parameters and demonstrates the Mandelbrot set, fractals, DE systems, and iterated maps (20). Moreover, other specific programs are in the literature. For example, Peitgen et al. provide a BASIC program that computes a particular strange attractor (23).

6.2 Dynamics Solver

As just mentioned, the downloadable freeware Dynamics Solver 1.01 appeared to be an excellent fit for this project (24). However, we could not install it on laboratory hardware, due to organizational network security restrictions. However, we describe it here in the hope that software restrictions may be lifted for the next phase of this project. (We opted not to pursue funding for other packages until the issues were resolved.)

Aguirregabiria developed the 32-bit edition of Dynamics Solver for Windows over several years. It handles initial- and boundary-value problems for continuous and discrete dynamic systems: single ordinary differential equations of arbitrary order, systems of any number of first-order ordinary differential equations, a large class of functional-differential equations and systems, and iterated maps and recurrences in arbitrary dimensions.

Generally, no programming is needed. User-friendly dialog boxes are used for entry. Numerical results and complex graphics may be easily obtained. A built-in compiler translates a large class of mathematical expressions written in a standard format for rapid execution. One can also compute quantities involving the solution and its derivatives, parameters, and initial conditions. It is possible to draw phase-space portraits, Lyapunov exponents, histograms, bifurcation diagrams, etc. Results may be projected and subspaces of the phase space or space of initial conditions easily analyzed. Any problem may be saved to and retrieved from an editable disk file. The program is configurable and extensible, and a complete help system is available.

Different kinds of results in graphics and text formats may be displayed in one or more windows. They may be sent to Windows-compatible devices or collected in a file for processing by other programs. Dynamics Solver may be used to construct many geometric figures, including parametric curves in two and three dimensions and a large class of fractals. The goal is to have completely correct figures in a device-independent format that can be translated for combination with output files from text processors.

Dynamics Solver is a powerful tool for engineers and mathematicians as well as a teaching device concerning dynamic systems. It is a reasonable numerical laboratory in which problems may be more readily analyzed and comprehended and explanatory graphics developed, in many cases without programming. Borrelli and Coleman is a useful companion volume to such efforts,

and we hope to utilize the program in such context in follow-on work (24). However, for this project, we were led to consider a contingency route.

6.3 Contingency

We have been developing an initial GUI by extending a Java-based plotting routine called JPlot (25). As an indication of the kind of work being done, we cite briefly in this subsection a few Java details; more extensive documentation will be released when the code is further along. For the first pass, we constructed the manner in which functions work as generation of an array of $(t, f(t))$ points. JPlot was written to plot data, not functions, and this appeared to be the quickest way to implement functions. Besides, a delta t should not be smaller than a pixel. There were many design considerations even in leveraging this existing software. For instance, when (if) the plot window is resized, should more points be computed, or should we always compute the same number of points based on the screen width? As for entering the functions, initially we preferred a list of canned functions for which the user provides the coefficients: $ax + b$, $ax^2 + bx + c$, etc. Eventually, there should be a mechanism by which a programmer could provide a function written in Java to avoid a clumsy (and complicated) translator.

We developed a working parser and evaluator for arbitrary expressions. However, for this initial version, the expression must be entered in postfix, or reverse Polish, notation with white space between the tokens. In other words, $2.3 + \sin(t)$ becomes “2.3 t sin +.” The infix form may be used for the description that the user sees. The initial version recognizes six binary operators ($^*/\%+-$), 10 unary functions (six trigonometric functions, exp, log, sqrt, abs), the constants pi and e, and the independent variable t. These have all been successfully tested and hooked into JPlot. After some experience with the software, it proved trivial to add new operators, such as sinh and cosh. We are considering the inclusion of neg (as in unary minus). We wanted to use Knuth's parser to convert infix to postfix, but volume 5 of *The Art of Computer Programming* will not be ready until 2010. However, given that the function can be parsed into a tree, volume 1 describes how to traverse the tree and build the postfix equation 25.

We have also made progress on when and how to build the DataArray, which in turn is plotted. We believe the function code has reached a beta level, despite current use of the JTable defaults. At this point, a plot of an array of data and a plot of a parsed function may be portrayed on the same graph. The user may define Reverse Polish Notation functions in a function file. The 24 letters of the alphabet, besides e and t, may be used as user-defined variables (named constants). The parametric range and increment may be supplied in the function file, as may the values of the named constants. When the user attempts to open a graph, the parametric values are checked to ensure they are defined and valid (e.g., $t_{min} < t_{max}$ and $dt > 0$). All variables referenced in the function(s) are also checked. At this point, it is possible a function that is not selected uses an undefined variable, which will prevent the graph from being generated, but generally there is a small chance of this actually happening.

We have added (conditionally) to PlotPanel a button labeled Vars to invoke an editVariables method in FunctionFile. It opens a JDialog containing two buttons and a JTable listing the names and values (if any) of all variables used in the current function file. The system ignores variables to which the user may have assigned values but did not use in a function. The Cancel button causes the changes to be ignored, while the OK button copies all of the values into a UserVariables array.

As an example of the GUI, the reader is referred to figure 4, which shows one prototype of the Jplot modification being used to input the square law model solution, and figure 5, which shows the resulting graph. This work is leading to a set of software that can link seamlessly with the main decision aid algorithms and enable interactive manipulation of the data, both in terms of input to the modeling and examination of the output. More JPlot updates are forthcoming, as well as a new application that permits use of sliders to adjust the variables. Sliders are easier (if more restrictive) than a relatively static Jtable for enabling the important requirement of “what if” analyses.

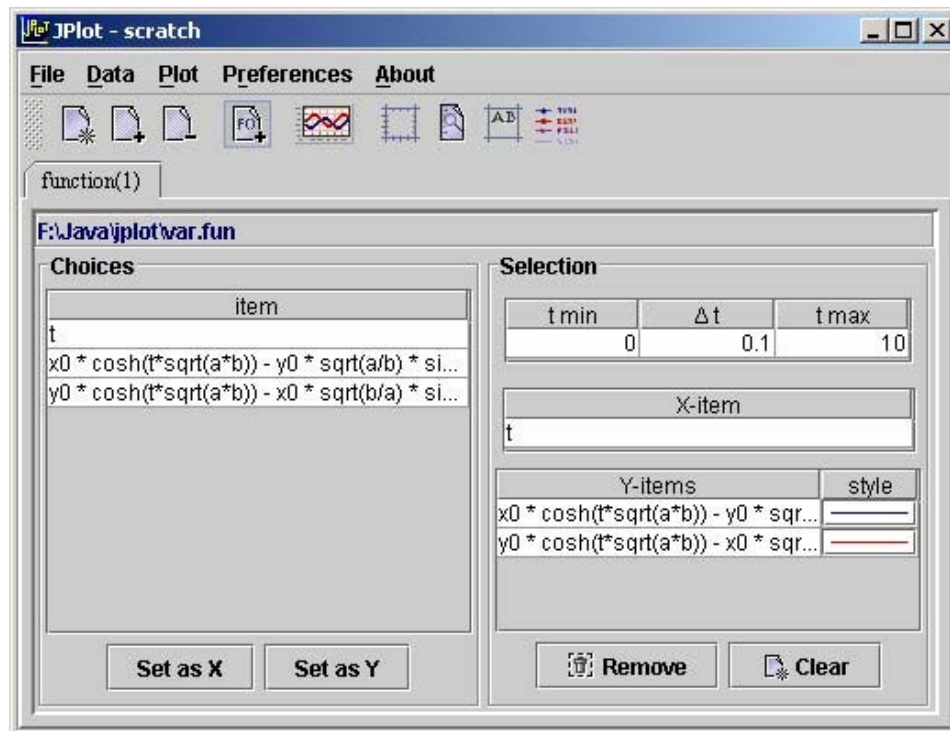


Figure 4. GUI input of the square law solution.



Figure 5. Graph resulting from GUI input.

In concluding this section, we note that since we could not gain authority to use the package of choice, we pursued different avenues. We did not delve into the experimental aspects of chaos and control, but rather looked more into design issues and setting the stage for follow-on mathematical and programming developments. In particular, we developed the parameter estimation aspects for the current model and Java plotting routines for the proposed GUI. We decided to go with the Java development, and to pursue use of DPGraph/Dynamics Solver and possibly Mathcad for future efforts. We have not given up completely on utilizing EnSight or PC MATLAB, although either of these may be overkill as total packages. In any event, we note that this software issue resulted in a change of plans in this project, and must be resolved to allow substantial effort along the lines originally considered. However, this was considered in a sense part of the research; moreover, the unexpected graphics development spin-off may lead to improved software for the Java community and a possible GUI patent. We also plan to pursue use of Dynamics Solver, DPGraph, and possibly Mathcad for future efforts.

7. Estimation of Parametric Values

7.1 Introduction

We now resume discussion of mathematical aspects of this project. As mentioned in the conceptual exposition, an important aspect of the prototype is that of improving the model on the fly. In order to do this, we assume that the battle may be monitored for estimated kill rate or,

more directly, estimated forces; an updated model of the battle may then be derived from the observations. As a particular application we chose to develop the parameters a and b from observations of troop strengths at several times. This led to interesting research into techniques, based initially on least squares curve fitting and difference equation approximations, with wider potential application.

If we assume force attrition on the battlefield obeys the Lanchester square law, then the solution is known and involves two parameters. Our original intent was to estimate these two parameters, given a set of force levels at various times during a battle. A recursive relation was found between the force level at a time and the forces levels at the previous two times. This relation allowed us to use a conventional least squares technique to determine the two parameters based on the available data.

The accuracy of future force levels is dependent on reported force strengths. Casualty reporting time intervals may depend on a number of factors, including commander's instructions, strength levels of the forces, and perhaps the importance of having timely strength data during the heat of battle. A battle expected to last only hours may require frequent casualty updates, while a battle expected to last days may not require as frequent an update cycle. Although accurate estimates of enemy force levels may be difficult to acquire and although attrition will, in general, not be as predicted by the calculated solution, smoothing techniques may be used on force levels to moderate the lack of precise knowledge, resulting in smaller changes in our estimates of the parameters. Smoothing techniques are used to better approximate the model solution, and an algorithm based on weighted moving averages and smoothed "derivative" estimates addresses a solution to distant estimates based on limited points.

Recall the discussion of kill (or attrition) rates in subsection 2.1. An extension to how commanders use their military judgment in assessing kill rates is beyond the scope of this report. However, many mathematical approaches are possible and are applied more or less consciously. For example, linearization is often used as a first approximation for estimating aspects of the battle. As a crude example, consider the following two observations: at time 0, $x(0) = 10$ and $y(0) = 100$; at time 10, $x(10) = 5$ and $y(10) = 90$. Now, the Blue loss rate over the interval is $(10-5)/(10-0) = 0.5$. Since these losses were inflicted by an average of $(100+90)/2 = 95$ Red, one argument is that the rate at which Red kills Blue is $0.5/95 = 0.005$. Similarly, the Blue kill rate can be computed as $([100-90]/[10-0])/([10+5]/2) = 0.13$. A more sophisticated technique we considered, and would like to develop further in future work, is based on the eigenvalue method of Anderson known as antipotential potential (27).

In any event, it would be desirable to present visual estimates of predicted troop strength levels to the commander along with the probable outcome of the battle. Toward this end we developed an algorithm to estimate force attrition levels based on reported casualty data. We made two important assumptions in this analysis: attrition obeys the Lanchester square law model as set forth by equation 1, and casualties are reported uniformly in time. Neither of these assumptions

will, in reality, be strictly true. However, to the degree they are true, some valuable insights into the progress of a battle may be gained.

The discussion that follows deals mainly with the strength levels of the Blue, or x , force; an analogous approach applies for the Red, or y , force. The general approach to predicting troop strength and thereby battle outcome will be the following:

1. Determine future troop strength recurrence relation based on previous troop strength as a function of a single variable c .
2. Estimate the value of this constant c based on the available casualty data.
3. Estimate the individual attrition rates a and b for forces x and y , respectively, based on the constant c and the available casualty data.
4. Based on initial troop strengths x_0 and y_0 and attrition rates $a, b > 0$, predict battle outcome and the effects of changes in the attrition rates or the probable outcome if additional or fewer troops are available.

7.2 Recurrence Relation Derivation

Given initial troop strengths x_0, y_0 for forces x and y , respectively, and attrition rates a and b , and assuming the force algorithm obeys the square law, then the solutions for x and y strengths at time t are, as noted in subsection 4.2,

$$x(t) = \frac{1}{2}[(x_0 - \sqrt{\frac{a}{b}}y_0)e^{\sqrt{ab}t} + (x_0 + \sqrt{\frac{a}{b}}y_0)e^{-\sqrt{ab}t}] \quad (43)$$

and

$$y(t) = \frac{1}{2}[(y_0 - \sqrt{\frac{b}{a}}x_0)e^{\sqrt{ab}t} + (y_0 + \sqrt{\frac{b}{a}}x_0)e^{-\sqrt{ab}t}]. \quad (44)$$

For this derivation, we are interested in integral numbers of time intervals and so replace t by an integer n , where it is understood that a time of n means n intervals.

A simple recurrence relation for $x(n)$ can be derived. Let

$$c = e^{\sqrt{ab}} + e^{-\sqrt{ab}}. \quad (45)$$

Strength at time

$$n-1 \text{ is } x(n-1) = \frac{1}{2}(c_1 e^{\sqrt{ab}(n-1)} + c_2 e^{-\sqrt{ab}(n-1)}), \quad (46)$$

where

$$c_1 = x_0 - \sqrt{\frac{a}{b}} y_0 \quad (47)$$

and

$$c_2 = x_0 + \sqrt{\frac{a}{b}} y_0. \quad (48)$$

Multiplication by c yields

$$cx(n-1) = \frac{1}{2} \left(c_1 e^{\sqrt{ab}n} + c_2 e^{-\sqrt{ab}n} + c_1 e^{\sqrt{ab}(n-2)} + c_2 e^{-\sqrt{ab}(n-2)} \right) = x(n) + x(n-2). \quad (49)$$

This finally implies

$$x(n) = cx(n-1) - x(n-2). \quad (50)$$

A similar derivation holds for y .

If we knew attrition obeys the square law, and if we knew $x(0)$, $x(1)$, $y(0)$, and $y(1)$ exactly, then it would be an easy matter to determine the value of c above and therefore predict all future strength levels for both forces exactly. These conditions are generally not satisfied, however, and so we must estimate the value of c given imprecise strength levels. Having a value for c does not require knowledge of individual attrition rates. In order to gain knowledge about the battle beyond simply predicting force levels, we estimate the individual attrition rates as constants based on the actual casualty figures.

7.3 Least Squares Approach

Given $N+2$ reported strength levels for x , $\{x_i \mid i=0,1,\dots,N+1\}$, we seek to determine a c value that minimizes the sum of the squared differences between the actual strength levels, as reported, and the predicted strength levels, as calculated. Let

$$S = \sum_{i=2}^{N+1} (cx_{i-1} - x_{i-2} - x_i)^2. \quad (51)$$

Minimizing S using the least squares approach will yield a particular value for c , say c_0 . Our interest in c_0 is that it will allow us to calculate \sqrt{ab} , since by definition in this case

$$S = e^{\sqrt{ab}} + e^{-\sqrt{ab}}. \quad (52)$$

Generally, in more complicated formulations (e.g., considering varying attrition rates as functions of time) we would have to solve analogous expressions numerically; however, in this case we can generate a closed form solution. If we multiply equation 34 by $e^{\sqrt{ab}}$ and let $z = e^{\sqrt{ab}}$, then we have the following quadratic equation:

$$z^2 - c_0 z + 1 = 0. \quad (53)$$

We solve this to obtain

$$z = \frac{c_0 \pm \sqrt{c_0^2 - 4}}{2} \Rightarrow \sqrt{ab} = \log_e \left(\frac{c_0 \pm \sqrt{c_0^2 - 4}}{2} \right). \quad (54)$$

By the nature of equation 34, we are interested only in the root

$$\sqrt{ab} = \log_e \left(\frac{c_0 + \sqrt{c_0^2 - 4}}{2} \right). \quad (55)$$

Note that the least squares process may give us a $c_0 < 2$. Since the range of the function $e^{\sqrt{ab}} + e^{-\sqrt{ab}} \geq 2$, this procedure will fail for $c_0 < 2$. In this case, a default value slightly > 2 is used.

Given a value for \sqrt{ab} , say α_0 , we seek to determine the value for the quantity $\sqrt{\frac{a}{b}}$. Using α_0 and rewriting equation 43, we obtain the equation

$$x(t) = \frac{x_0}{2}(e^{\alpha_0 t} + e^{-\alpha_0 t}) + \frac{y_0}{2}(-e^{\alpha_0 t} + e^{-\alpha_0 t})\sqrt{\frac{a}{b}}. \quad (56)$$

Let

$$S = \sum_{i=2}^{N+1} \left(\frac{x_0}{2}(e^{\alpha_0 t} + e^{-\alpha_0 t}) + \frac{y_0}{2}(-e^{\alpha_0 t} + e^{-\alpha_0 t})\sqrt{\frac{a}{b}} - x_i \right)^2. \quad (57)$$

We seek to minimize S , which will give us the value for $\sqrt{\frac{a}{b}}$ that minimizes the sum of the squared differences between the actual and predicted x strength levels. Suppose the value of $\sqrt{\frac{a}{b}}$ is α_1 .

We now have the following two equations: $\sqrt{ab} = \alpha_0$ and $\sqrt{\frac{a}{b}} = \alpha_1$. Solving this system yields $a = \alpha_0 \alpha_1$ and $b = \alpha_0 / \alpha_1$. These values for a and b are then, in the least squares sense, the best estimates we can make given the realization that reported casualty figures will not strictly adhere to the square law.

7.4 Modifications

The force strength curve in a real battle would more likely resemble a step function having constant strength over a number of time intervals. In an effort to make the reported strength curve more closely resemble the curve one would get if attrition obeyed the square law, we modify the originally reported data and generate two additional sets of data.

The first modification uses a weighted moving three-point average. Currently, we use $x_i^s = (x_{i-1} + 2x_i + x_{i+1})/4, i = 1, 2, \dots, N$. The superscript denotes that these values have been smoothed.

The second modification is more complicated. A force whose attrition obeys the square law has, in addition to a “smooth” strength curve, a “smooth” rate of loss (i.e., first derivative) curve. In a real battle, the loss differences would likely vary wildly as a function of time. Again, in an effort to force the loss difference curve to more closely resemble the curve one would get if attrition did obey the square law, we fit a quadratic polynomial to the loss differences. In this scheme, we replace the actual loss differences with ones calculated based on quadratics B and F fitted to the backward and forward differences, respectively. To modify the smoothed point x_n^s , we set

$$x_i^D = (x_{i-1}^s + x_i^B + 2x_i^s + x_{i+1}^s - x_i^F)/4, \quad (58)$$

where x_i^B and x_i^F are the quadratic-generated derivatives at the i th time interval using the backward and forward differences, respectively. The intent is to take into account the fact that nearby points are changing smoothly and predictably in developing a process that will result in reasonable strength values. We place two restrictions on the derivatives generated: all derivatives are less than zero and all derivatives are strictly decreasing. In the event either of these conditions is violated, the terms x_i^B and x_i^F are ignored in equation 9, which then reduces to a weighted moving average of smoothed points.

For each of the three data sets, we generate c values minimizing the squared errors between reported and predicted x levels, similarly for the y levels, and similarly for both x and y simultaneously. For each of the three \sqrt{ab} , values then calculated for each set, we calculate the $\sqrt{a/b}$ value minimizing the squared errors for the x force and similarly for y . For each of the 18 combinations of \sqrt{ab} and $\sqrt{a/b}$ we calculate a and b . We rank by size the sum of the squared errors, subtract the smallest value from each, and eliminate those beyond a selected maximum allowable value E_{MAX} . (As an alternative, we ranked errors by the sum of the absolute differences, but found such ranking did not change significantly; its effect on the calculation of an average a and b was negligible.) The rest are weighted by

$$w_i(a_i, b_i) = \left(\frac{E_{MAX} - E_i}{\Delta} \right)^K, \quad (59)$$

where E_i is the actual sum of squared errors, Δ is the partition size, and K is an exponent used to change the distribution of the weighting scheme. The attrition coefficients used for

characterizing the reported strength data are then $\hat{a} = \frac{\sum_{i=1}^{N_w} a_i w_i}{\sum_{i=1}^{N_w} w_i}$ and $\hat{b} = \frac{\sum_{i=1}^{N_w} b_i w_i}{\sum_{i=1}^{N_w} w_i}$, where N_w is the

actual number of a, b combinations used in the final determination.

A little more discussion on E_{MAX} is warranted. This value is basically a function of the number of points used in the calculation of the sum of squared errors, and increases with the number of points. For example, if we believe all errors >1000 would indicate a “bad fit” and we chose $\Delta=100$ and $K=1$ (i.e., linear weighting), then we generate weights in $[0, 10]$. A value of $K=2$ increases the range of values to $[0,100]$ and has the effect of giving increasing weight to those a ’s and b ’s having smaller errors.

The final values used to characterize the reported strength data are then $\hat{a} = \frac{\sum_{i=1}^{N_w} a_i}{\sum_{i=1}^{N_w} w_i}$ and $\hat{b} = \frac{\sum_{i=1}^{N_w} b_i}{\sum_{i=1}^{N_w} w_i}$, where N_w is the actual number of combinations used in the final determination.

7.5 Illustrations

The following paragraphs describe plots illustrating some of the results of this work. The plot in figure 6 shows the predicted strength using 10 and 20 reported strength levels when force strength obeys the square law. The strength prediction was good and tells us that if casualty figures are reported on time and are accurate, then a good estimate of future strength will be derived. Attrition rates are $a = 0.01$, $b = 0.02$.

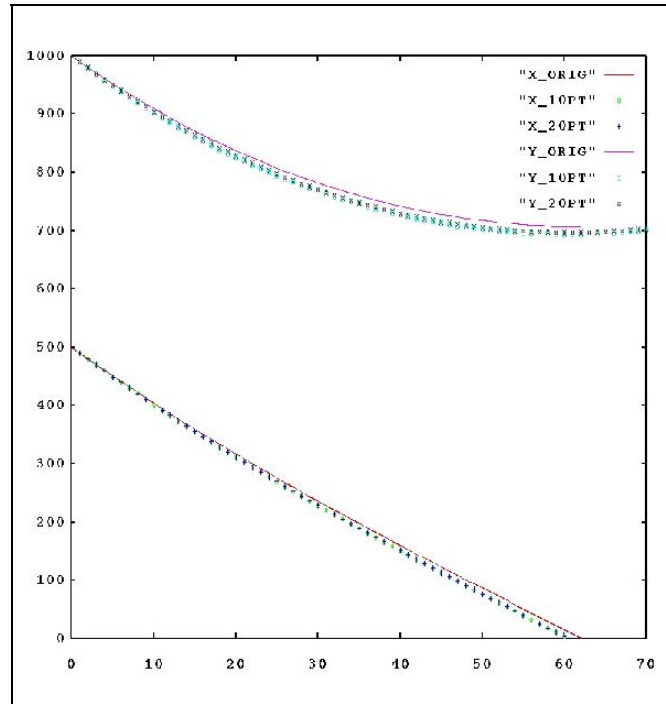


Figure 6. Predicted strength.

We calculate a smoothed estimate of the difference in force levels for each time interval (figure 7). XD_EXACT represents the differences one would expect if x strength data were generated exactly as a square law model. XD_ORIG represents the difference after the exact strength data was modified to reflect some delay in reporting strength data resulting in constant strength over several time intervals. We replace each calculated difference by the value of $A_0 e^{Bt}$, for some constant A_0 nominally set to a weighted average of the first few values and for B which minimizes the errors between XD_ORIG and predicted values generated by the function $A_0 e^{Bt}$ in either the least squares sense or the sum of absolute errors sense. In this instance, the absolute value approach more closely approximated XD_EXACT, although as discussed this was generally not the case. The upper and lower sets of curves used 10 and 20 points, respectively.

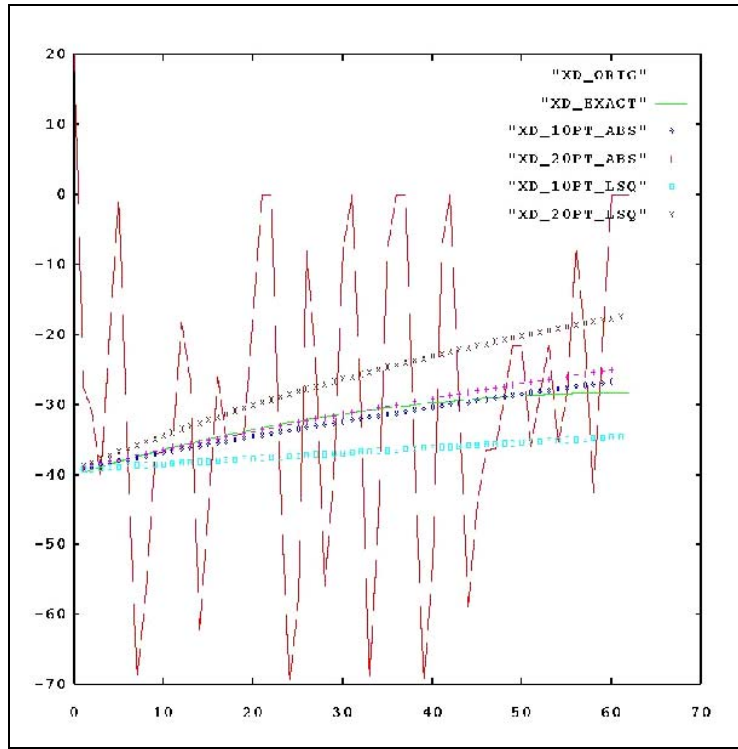


Figure 7. Smoothed estimate.

The plot in figure 8 shows an example of y force strength levels used by the algorithm. The exact trace of a square law generation is shown as Y_EXACT. Y_ORIG is derived from Y_EXACT by assuming that several time intervals may elapse before casualties are reported. Y_DERIV is generated by averaging a smoothed value y_i with its previous value y_{i-1} and next value y_{i+1} projected one time unit forward and backward, respectively, by utilizing our estimates of the smoothed derivatives at those points.

Given x and y force strength data X_ORIG and Y_ORIG , we plot the predicted force levels, using the algorithm, based on 10-point and 20-point reported strength levels (figure 9). We believe these are good predictions. Attrition rates are $a = .01$, $b = .02$.

The plot in figure 10 shows a situation in which x initially has 600 troops and y has 500 troops, but attrition rates are such that if the battle proceeds without change, then x will lose in 88 time intervals. At the fiftieth interval, x and y strengths are 210 and 300, respectively. At this point, x receives an additional 240 troops, which allows x to withstand enough losses and inflict enough casualties to win.

The plot in figure 11 illustrates the quantities used in determining the attrition rates for the two forces. The 18 a and b values are plotted, along with their weights, for each of the ranked combinations, as well as the final $a_average$ and $b_average$. Conditions were: $x = 2000$, $y = 4000$; $a = 0.01$, $b = 0.02$; casualty reporting delayed by a random number of intervals.

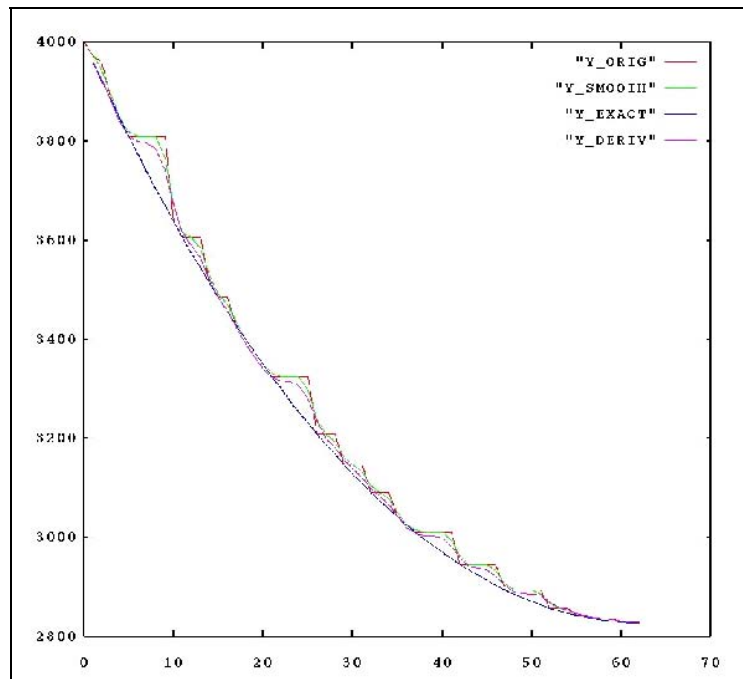


Figure 8. Y-force strength levels.

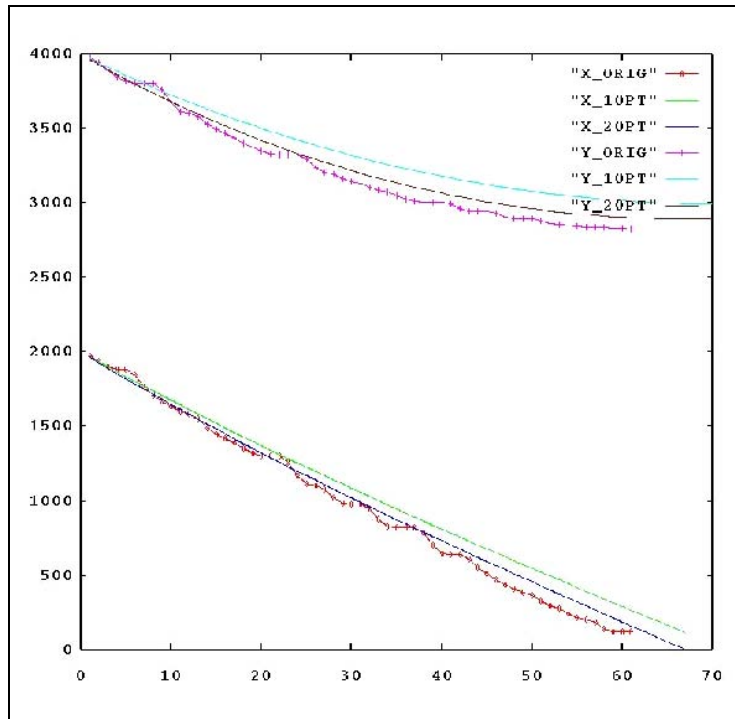


Figure 9. Predicted strength levels.

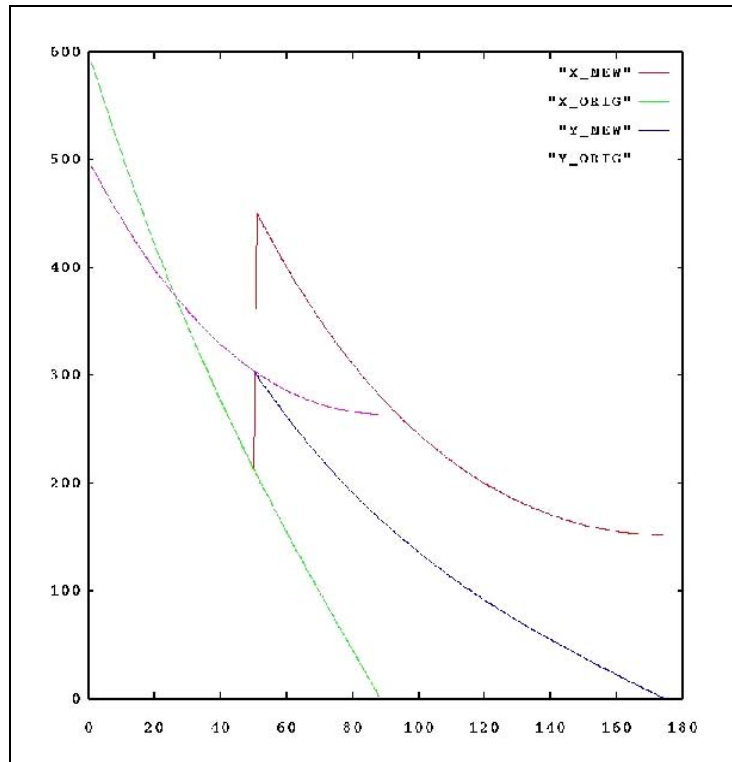


Figure 10. Modified situation.

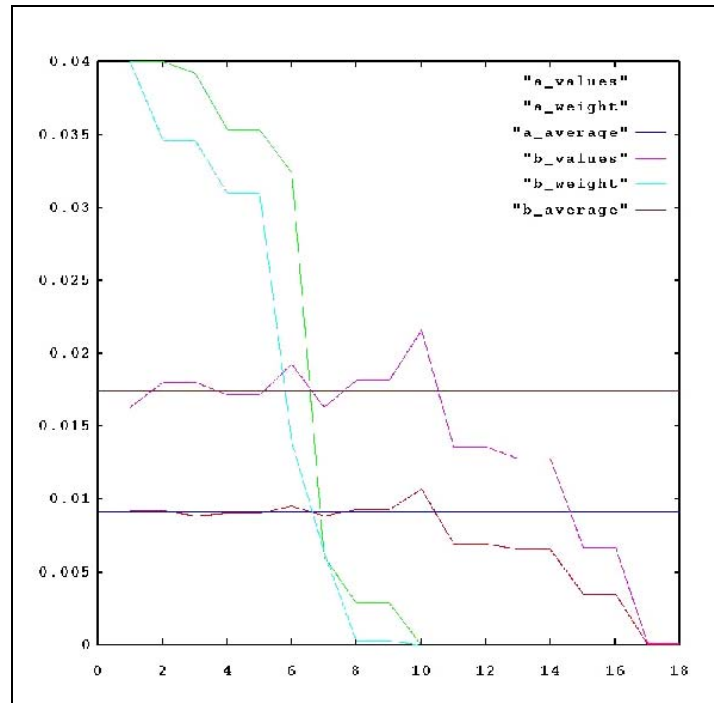


Figure 11. Attrition rate quantities.

Figure 12 shows predicted levels of Iraqi forces in Desert Storm based on 5, 10, and 15-day assumed casualties. Daily reports were not available, and the following assumptions were made: coalition forces, 360,000; Iraqi forces, 540,000; coalition casualties, 358; Iraqi casualties, 100,000; duration, 43 days (28). Attrition rates were chosen so that after 43 days roughly 100,000 Iraqi troops had been killed. We calculated force levels assuming a square law and then added a random number between -5000 and 5000 to reflect uncertainty of coalition estimates of Iraqi casualties. Agreement between predicted and assumed force levels is quite good. Note that the actual daily casualty figures may be vastly different than what is shown and that these force levels reflect a constant attrition rate over the duration of the war. The agreement between predicted force level and the assumed force levels was quite good.

Figure 13, based on the same scenarios as in figure 12, reflects the situation where Iraqi casualty reports were delayed by a randomly selected number of days (uniform distribution between 0 and 5).

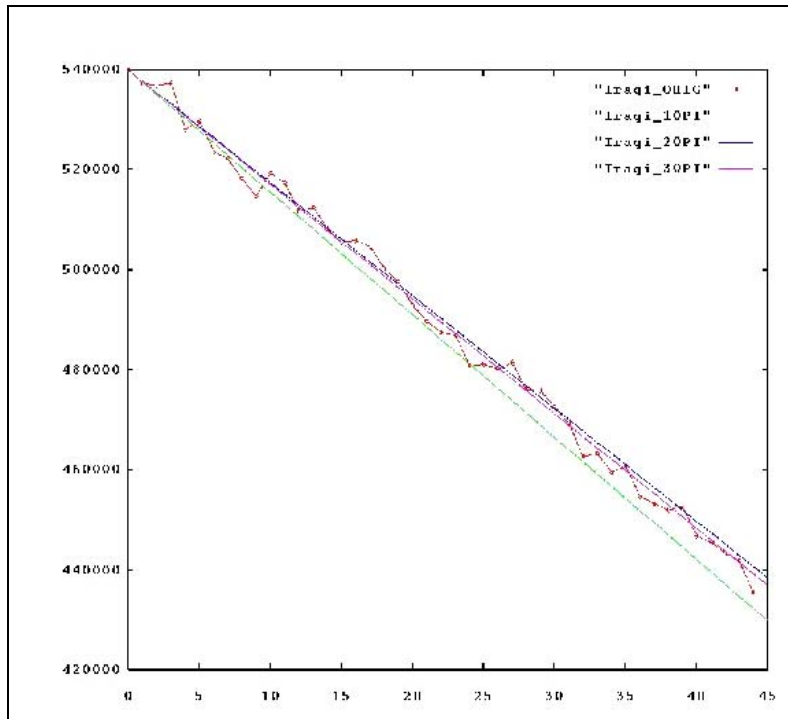


Figure 12. Predicted Iraqi forces.

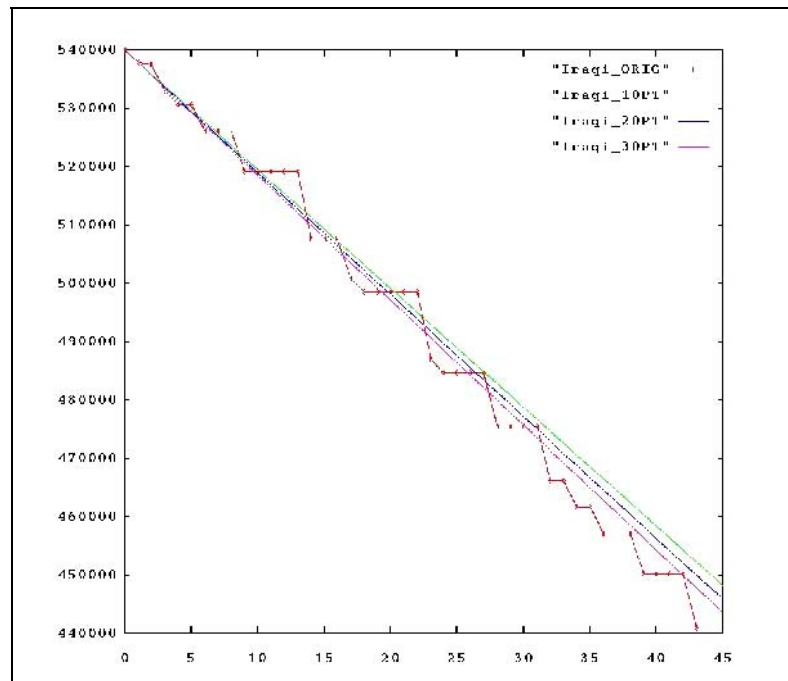


Figure 13. Delayed casualty reports.

8. Techniques for Analysis of Dynamic Systems

8.1 Introduction

We now shift gears a bit to discuss some germane concepts that are being used in ongoing work to develop the decision aid with regard to qualitative analyses of the system model. As mentioned earlier, we are seeking application of results in nonlinear dynamics (a portion of which is sometimes known as mathematical “chaos” theory) to COA problems. Philosophical expositions concerning sensitivity to small variations in initial conditions have been valuable for raising the consciousness of the networks and information integration (NII) community. However, we seek rigor in treating combat as a dynamic system. For instance, we are concerned about defining the senses in which combat can be treated in terms of iterative function theory. We are also interested in applications of optimal control theory (observability, controllability, and stability), particularly since such investigations may ultimately lead to tools for monitoring and even correcting divergence from a plan.

Any analyst, when confronted with understanding or designing a system, will consider certain questions. What are the parameters? Can they be measured and controlled? Does the system exhibit nonlinearities or feedback? Is it repetitive? Is the phase space bounded? Do the phase variables mix? Particular technical challenges may be expressed in terms of two “simple” questions. Does so much variation exist in combat, given even nominally similar plans and scenarios, that there is no way to group them into “bins” of similar trajectory in phase space? Do natural ways exist to develop envelopes, forbidden regions, or attractors that permit assessment of the influences of various combat factors? A related consideration is that of assessing confidence in the outcome and, by extension, actual sensitivity analyses of chaotic combat situations.

Difficult problems abound in qualitative investigations of nonlinear systems. For some, no general method has been developed (29); for instance, the existence of integral curves of certain kinds not restricted to lie in the neighborhood of a certain singular point, and limit cycles characterizing the disposition of integral curves in the large. Thus, these are worthwhile investigations, even in the realm of pure mathematics, and they should be somewhat more readily related to when applied to specific systems as we are doing.

Several commercial or academic software packages designed for “chaos exploration” or “fractal exploration” exist and are described briefly in section 6. We intend to utilize such software in conjunction with the other commercial off-the-shelf packages and experimental software for further research and development of both the theoretical underpinnings and the actual decision aid prototype. Along these lines, we note another germane use for the large OneSAF/DISAF databases mentioned elsewhere in this report. The results of executing the stochastic simulation

based on the same COA inputs can be plotted in different kinds of phase space and analyzed for trajectory “bins” and the existence of attractors. This sort of work is of interest to the U.S. Army Battle Command Battle Laboratory and holds much promise for valuable insights into the nature of at least certain kinds of combat.

We are interested in graphically portraying the behavior of continuous-time dynamic systems; in particular, limit sets, basins of attraction, and trajectories. Initially, we consider only second-order systems, because all possible types of behavior can be displayed in the plane. Moreover, we intend to examine only structurally stable systems due to numerical analytical considerations. Such notions will be discussed shortly.

8.2 Theoretical Notions

Czerwinski has noted, “By post-Newtonian, we mean the arrangement of nature—life and its complications, such as warfare—to be nonlinear, where inputs and outputs are not proportional; where phenomena are unpredictable, but within bounds are self-organizing; where unpredictability frustrates conventional planning; where solution as reorganization defeats control as we think of it; and where a premium is placed on nonlinear reductions. And where rewards go to those who excel in coping with the bounds in order to command and manage—not on prediction and control” (30). Also, Weaver suggested three classifications of complexity: “organized simplicity,” described by a small number of parameters deterministically linked, but some of which can be neglected (and which we generally analyze through mathematical modeling); “disorganized complexity,” in which a large number of distinct nondeterministic parameters play the main role (and which we tend to describe statistically); and “organized complexity,” in which many interrelated but deterministic parameters all contribute to the system (31). We hope further examinations of chaos theory will help us understand the third type, of which combat is an example. With these observations as motivation, we sketch out some concepts that will prove helpful in subsequent decision aid development.

A dynamic system is a way to describe how one state in phase space evolves into another over time. A dynamic system is a smooth action of the real numbers (a continuous system) or the integers (a discrete system) on another object. If f is a continuous function, the evolution of a variable x can be given by the formula $x_{n+1} = f(x_n)$. This equation can be transformed into a difference equation that forms the discrete analog of the differential equation $\dot{x} = f(x)$, and the difference equation formulation provides some incentive for the solution to \hat{a} and \hat{b} set forth in section 7.

A set of points in phase space is said to be an attractor if it is invariant under the dynamics and certain neighboring points asymptotically approach the set. In particular, an attractor upon which trajectories are periodic is said to be a limit cycle; trajectories circle around a limiting trajectory which they asymptotically approach. A set of points in the space of system variables such that initial conditions chosen in this set evolve to a particular attractor is said to be a basin

of attraction. An attractor is the smallest unit which cannot be decomposed into more attractors with distinct basins, a distinction needed because a system may have several attractors, each with its own basin.

Moreover, in a system with multiple basins of attraction, the boundaries may be in a fractal pattern in which nearby initial conditions lead to radically different trajectories. Decision making in such environments will probably be difficult without a detailed aid. James makes this point concerning a decision space formed by the number of reinforcement troops available and the time intervals between reinforcements (32). An interesting question is whether in the tactical domain we see fractal boundaries of the basins. If some trajectories within some epsilon of an initial point converge to one attractor and some to another, this point is said to be “unsafe” (23); one cannot safely predict the final state associated with the initial point. Moreover, fractal boundaries with a large dimension further hinder predictability in such nonlinear systems. It is obviously important to understand these aspects of combat, if only to try to “know what one does not know” or place limits on the reasonableness of attempting to observe or control such systems.

Nicolis and Prigogine point out that in systems having dynamics reducible to a 2-D phase space the only attractors are limit cycles and fixed points (33). With additional dimensions we find more complicated topologies of trajectories around these types of attractors, as well as new types of behaviors. One way to explore these is through studying the succession of points in a plane that cuts the trajectories; by using the so-called “Poincaré map,” we derive information about the attractor.

8.3 Chaos

A system described by a function f is said to be chaotic if it (1) is sensitive to initial conditions (that is, initially neighboring states can evolve to distant states), (2) has a dense set of states with periodic trajectories, and (3) is such that, given any two open sets U and V , a positive integer n exists such that the intersection of $f^n(U)$ and V is not empty (that is, neighborhoods of points get flung out eventually to “big” sets) (34). The first item is the one most popularly associated with chaotic systems, “unpredictable” behavior. Simple physical examples of chaotic systems are a pendulum with another attached to the free end and a magnetic pendulum moving over a plane containing two attractive magnets. Motion and end state of both systems are highly dependent on initial position and velocity.

A chaotic system must be bounded, nonlinear, nonperiodic, sensitive to small disturbances, and mixing (in the sense just discussed). James believes these criteria are necessary and sufficient (32). Also, such a system usually has these observable features: transient and limit dynamics, parameters, definite transitions to and from chaotic behavior, and attractors. In 1976, Rössler discovered what is probably the most elementary geometric chaotic system:

$\{\dot{x} = -(y + z), \dot{y} = x + ay, \dot{z} = b + xz - cz\}$ (23). This is an artificial formulation to create a

strange attractor using the simple stretch and fold chaos generator. We are attempting to develop an analogously simple system with rationale in the tactical realm.

Several techniques, such as Greene's method and resonance overlap, exist for predicting the onset of chaos in a system as parameters are adjusted. However, these are beyond the scope of this initial exposition. Again, follow-on efforts should address the analysis of orbital transition. Bradley has developed a (computationally intensive) software approach to controlling chaos in electrical circuits, by using information about dynamics on the attractor to control individual trajectories to a target point in phase space (35). A few years ago, this was also being done at ARL in the context of communications (36).

Some detailed mathematical expositions have been done in the combat modeling community with regard to chaotic behavior. For example, Oak Ridge National Laboratory (ORNL) has done work on DE-based models (37) exhibiting chaos, and RAND has shown chaos arising from certain models, even very simple ones (38). In particular, Dockery and Woodcock have analyzed several models, including Lanchester-based and ones incorporating reinforcement, utilizing Lyapunov exponents (discussed in the next subsection) and fractal mathematics (39).

8.4 Stability

In studying systems (linear or nonlinear), one generally tries to characterize them by looking for equilibria, seeing if they are stable, estimating growth rates, considering behavior resulting from perturbation, considering limiting behavior, and so forth as we have been discussing. We are concerned about stability in the face of changing parameters, asymptotic behavior of solutions, and periodic or chaotic behavior.

Qualitative properties can be developed without explicit solutions, which are generally difficult. Portraits, usually via numerical solution, provide information about the general behavior. Time-space curves are another tool. Of course, in high dimensions we have the usual visualization problems, and may turn to component graphs or other projections of the full state space onto a subspace.

Observe that dy/dx gives the slope of the trajectory that passes through the point (x, y) . If the numerator and denominator are both zero, the point is a critical or equilibrium point of the system; if there is a circle around (x, y) that contains no other critical point, it is said to be isolated. The isolated critical point is stable if and only if for every distance ε there is a distance δ such that any trajectory that comes within δ of the point remains within ε of the point subsequently. The point is said to be asymptotically stable if and only if it is stable and every trajectory that comes sufficiently close to the point actually approaches the point.

Wiggins notes that it would be useful if structural stability (where, roughly speaking, this means "nearby" systems have "qualitatively the same" dynamics) of a system could be characterized, if only since it might be presumed models of natural phenomena should be structurally stable (40). Unfortunately, such determination does not exist (although partial results are known). However,

Wiggins discusses at length one approach to such characterization, namely identification of generic properties of dynamic systems. It would be interesting to know whether certain models are structurally stable since, by definition, perturbations of unstable dynamic systems can produce systems having radically different behavior. Again, such knowledge would be valuable to the commander; however, the mathematical development lies in the future.

Understanding the nature of possible configurations of trajectories of simple linear systems is essential for the descriptive study of general systems. If an entry in the system matrix of a linear system is modified, so are its trajectories, and (as just noted) structures vary in sensitivity to parameter changes. The stability of (0,0) for the system $\{\dot{x} = ax + by, \dot{y} = cx + dy\}$ (where a, b, c , and d are constants) can be examined via the characteristic equation (the determinant of the operational coefficients equated to zero),

$$m^2 - (a + d)m + (ad - bc) = 0 \quad (41). \quad (60)$$

Borrelli and Coleman show how to use eigenanalysis to construct the solution of a homogeneous linear system with constant coefficients. They also classify all portraits for a real 2×2 matrix, based on characteristics of the eigenvalues: improper node, deficient improper, star, saddle, vortex, and spiral (13). Table 2 gives stability properties of the critical point (0,0) of the linear system just mentioned.

Table 2. Stability of (0,0) for $\{x' = ax + by, y' = cx + dy\}$.

Nature of the Characteristic Roots	Nature of the Critical Point	Stability of the Critical Point
Real, unequal, of like sign	Node (improper)	Asymptotically stable if roots -; unstable if +
Real, unequal, of unlike sign	Saddle point	Unstable
Real and equal	Node (proper or improper)	Asymptotically stable if roots -; unstable if +
Pure imaginary	Center	Stable, but not asymptotically stable
Complex but not pure imaginary	Spiral point	Asymptotically stable if real part of roots -; unstable if real part +

In particular, expressing the square law equations in the form $\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t)$ results in the

system matrix $\mathbf{A} = \begin{bmatrix} 0 & -a \\ -b & 0 \end{bmatrix}$. Hence, the characteristic equation for our initial model

is $m^2 - ab = 0$, with roots $\pm\sqrt{ab}$, and therefore the origin is an unstable saddle point. This is also observable in earlier illustrations of the tactically significant quadrant (that is, where x and y are both non-negative) of the larger (x, y) -plane plot of the system.

As just alluded, if the system is linear, the signs of the real parts of the eigenvalues generally determine the stability properties. Different tools exist for nonlinear systems and one might also

look for threshold effects or other aspects. The most useful tool for nonlinear systems is the summarizing function of the system state vector, and it is often possible to describe its behavior by an approximate first-order DE. Analyzing that equation then summarizes the system. The summarizing functions used for examining the stability of nonlinear systems are called Lyapunov functions.

In the method of Lyapunov, stability of complex systems is generally measured in terms of “distances” of several evolving states from an equilibrium state and determining whether these distances decrease, or by measuring the energies of evolving states and determining whether they decline (13). Lyapunov exponents generalize the notion of eigenvalues at points of equilibrium, and can be used to investigate such points, periodic, quasiperiodic and chaotic behavior. We can assess the motion’s instability through the use of the mean exponential rate at which two initially close trajectories diverge. The Lyapunov exponents are the projections along the n independent directions in phase space of this mean rate:

$$\lambda_t(\mathbf{X}_0, \Delta\mathbf{x}_0) = \lim_{t \rightarrow \infty} \frac{1}{t} \log_e \frac{|\Delta\mathbf{x}(\mathbf{X}_0, t)|}{|\Delta\mathbf{x}_0|}, \quad (61)$$

where we are considering two trajectories having initial conditions \mathbf{X}_0 and $\mathbf{X}_0 + \Delta\mathbf{x}_0$ and the limit is taken $t \rightarrow \infty$ and $|\Delta\mathbf{x}_0| \rightarrow 0$ (33).

By measuring the sensitivity of trajectories to perturbation, Lyapunov exponents provide information that can help the commander find battle states in which a plan is less sensitive to disturbance, for instance, of regions of parameter space in which the combat variables may produce less predictable outcomes. We will be examining such calculations in follow-on work, possibly utilizing Raglin’s expertise garnered in the image processing realm (42).

Elementary catastrophe theory deals with equilibria and families of potentials and how these equilibria change when external control parameters change. However, it cannot describe the dynamic processes in transitioning between static equilibria. Moreover, it is difficult to extend its results to dynamic systems, which have no analogous canonical forms, although certain methods of phase portrait analysis can be used to facilitate analysis of some dynamic systems. Qualitative topological methods can provide much information with relatively little effort. Also, catastrophe theory may be applied to problems (such as combat, arguably) in which neither the variables nor equations governing the system are known. Given limited modality, a small number of functions of state variables actually drive the system, and qualitative behavior in the neighborhood of certain critical points may be reasonably inferred (43).

Understanding the nature of the manifold upon which the battle trajectories play out would shed more light on possibilities for predicting outcomes. Parker and Chua set forth algorithms useful for finding stable and unstable (one-dimensional) manifolds of a second-order system, and note that the algorithms can also be applied to higher-order systems (21). They also note that

techniques for finding higher-dimensional manifolds are not known. Mathematical analysis of the phase space itself is a ripe area of research. Understanding the nature of the manifold upon which the battle trajectories play out would shed more light on possibilities for predicting battle outcomes. Notions of conformal mapping and area/volume preservation in phase space may seem far removed from actual battle, but investigation of such topics may have its place in development of future military decision aids.

8.5 Deviation From Plan

As we have discussed, it is important for the commander to appreciate when the events of the actual battle are causing, or probably will cause, significant departure from the projected trajectory of the conflict as modeled based on his input plans. Of course, there are many simplistic approaches such as just tracking the Blue casualties and alerting the commander whenever the number exceeds the postulated number. However, it would be much more valuable to develop meaningful measures of departure, in a spirit of understanding the larger dynamics of the conflict. The fact that we have vectors of parameters characterizing the battle makes the mathematics interesting and challenging.

Certain approaches have been discussed in section 5. Lyapunov analysis is another possibility. Divergence measures could be based on schematics like

$$\begin{cases} P(t) = P(0) + \int_0^t v(t)dt \\ \tilde{P}(t) = P(0) + \int_0^t \tilde{v}(t)dt . \\ \Delta(t) = \int_0^t (v - \tilde{v})dt \end{cases} \quad (62)$$

In such a formulation, we could consider the actuality to be critically deviant from the plan if Δ exceeds some threshold. But here again the rub is how to choose the threshold in a meaningful, dynamic, tractable, and “situation-dependent” manner. One concept worth considering is that the divergence is too great if the model shows the commander will be unable to bring the trajectory back to the desired path within some desired time. This sort of “derivative” modeling has been discussed earlier, and forms to a large extent the motivation for the control theoretical aspects being pursued.

We cannot (just) consider time in the analysis, since even relatively simple “phase shifts,” as opposed to the “amplitude differences” generally analyzed, could cause apparently dramatic departures; we must consider the plan vis-à-vis enemy strength, friendly position, and other parameters that are in some senses independent of the temporal flow of the battle. Moreover, both the time-stamped and the event-based indicators are subject to all the uncertainty and error

associated with the fog of war. As indicated in section 7 on parameter estimation, an approach dealing with assumed certain updates is challenging enough initially.

We will be exploring divergence criteria in more depth in follow-on work. Since we have in effect n -dimensional pulsating clusters as a function of time, we hope to be able to apply some of Liao's methodology (10, 44). Further, since time can serve as a "dependent" variable as well, we must consider notions of critical deltas and combined deltas.

8.6 Control Theory

We remind the reader that the idea that some aspects of the conflict are unknown to the commander is an important aspect of this project. With this in mind, we sketch some notions that should be quite important in future developments. If we want to consider the existence of a solution to designing an optimal system, we must investigate whether its state is "observable" and whether it is "controllable." Usually, control theoreticians attempt an optimal solution to a well-defined, modifiable system. However, the types of model we have been considering are in certain senses unmodifiable. Moreover, we have found in other related investigations that it is difficult to develop an objective function for combat (3). These aspects of the system probably make calculation of observability and controllability more worthwhile than that of an optimal solution, which may not even exist (45). It is hoped that analytical investigations into observability, controllability, and stability will yield improved understanding of the dynamics of the conflict.

As noted earlier, our continuous time system can be expressed as $\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t)$. The so-called state transition matrix, by which multiplication of an initial state vector yields the state at a future time, can be calculated from $e^{\mathbf{A}(t-\tau)}$, for our time-invariant case. The details of this development are beyond the scope of this exposition; however, we note certain parallels in the recurrence relation derivation of subsection 7.2 (46).

Now, the general form of the state space equations of a lumped-parameter, linear, continuous-time system is

$$\{ \dot{\mathbf{x}} = \mathbf{A}(t)\mathbf{x} + \mathbf{B}(t)\mathbf{u}(t), \mathbf{y}(t) = \mathbf{C}(t)\mathbf{x} + \mathbf{D}(t)\mathbf{u}(t) \}. \quad (63)$$

Typically, the control problem comprises determining what the input $\mathbf{u}(t)$ should be so that the state $\mathbf{x}(t)$ or the output $\mathbf{y}(t)$, or both, behave acceptably as the goal of the system is achieved.

Controllability, a property of the coupling between input and state, involves \mathbf{A} and \mathbf{B} ; observability, a property of the coupling between state and output, involves \mathbf{A} and \mathbf{C} (47).

A system is said to be completely observable if by observing the outputs we can deduce the initial state in finite time. Many control designs are based on the assumption that the state vector can be measured completely all the time; the current input is specified as a function of the current value of the state vector. Intelligent control should be based on the current state. However, in many systems (in particular our combat model) the entire state vector is not

available for measurement (for a variety of reasons), and so control of such systems must be based on a subset of the state variables. One could deal with this by looking for techniques that require fewer measurements or by constructing an approximation to the full state vector based on measurements that are available. It can be shown that an approximation to the state can be computed by another linear dynamic system whose state vector generates missing information about the state of the system being studied (46).

The system $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t)$ is said to be completely controllable if for $\mathbf{x}(0) = \mathbf{0}$ and any given \mathbf{x}_1 there is a finite t_1 and a piecewise continuous input $\{\mathbf{u}(t), t \in [0, t_1]\}$ such that $\mathbf{x}(t_1) = \mathbf{x}_1$. It can be shown that a continuous-time system is completely controllable if and only if the controllability matrix $\mathbf{M} = [\mathbf{B}, \mathbf{A}\mathbf{B}, \mathbf{A}^2\mathbf{B}, \dots, \mathbf{A}^{n-1}\mathbf{B}]$ is of rank n (46). Again, the details of this exposition are beyond the scope of this report. However, these sorts of analyses will play a significant role in development of sophisticated decision aids that incorporate the notion of controlling the battlespace based on observations of the unfolding conflict, particularly if appropriate models turn out to be highly nonlinear, as appears likely.

Analytical solutions to systems of nonlinear equations are generally not possible. However, there are techniques that can obtain approximate solutions, yielding understanding of the behavior of a system when perturbed from a known solution and developing iterative corrections often required when controlling such solutions (47).

9. Future Efforts

9.1 Extension of Initial Model

Many extensions are possible, to varying levels of difficulty, to make our simple model less abstract and bring it more into line with tactical reality. For instance, it is well known from historical battles that unit “posture” affects kill rates on both sides. Incorporating this “simple” change into the Lanchester formulation results in another degree of internal self-reference and complexity. Similarly, attrition factors are probably dependent on time, as manifested in terms of fatigue and destruction of equipment if nothing else. Change in killing power could be due to Blue improved/worsened weapons, Red worsened/improved defense, or changing distance. Other changes to the system of differential equations modeling the conflict would be relatively straightforward, at least in a theoretical sense if not a tactical. “Reinforcement” of one or both force levels could be brought about by some function; it could be negative, like disease or desertion. Change in breakpoint could be due to morale or commander’s guidance. Any or all of these factors could be considered to be changing in an essentially continuous manner.

Explicit consideration of troop movements complicates the solution immediately (even if handled at a high level of abstraction), as will be discussed in subsection 9.3. At the least, however, we must consider reachability of phase lines and achievement of intermediate goals.

More explicit treatments of subconflicts involving split forces and of phases of battle are desirable extensions with great analytic power. Work was initiated in this project along these lines, as we considered reinforcement at time τ , optimal allocation, Blue engaging two Red forces, and effects of delay on the assault. However, much remains to be done; we plan to report on such developments in future reports. A more complicated improvement would be inclusion of disparate groups of forces in a multisided game, analogously to that being pursued by other ARL researchers (4). Again, initial mathematical results are setting the groundwork for programming algorithms to improve the prototype.

9.2 Attrition

Taylor notes several functional forms for attrition rates. For instance, Lanchester himself considered the effect of firer and target strength on both sides in the formulation

$$\left\{ \frac{dx}{dt} = -axy, \frac{dy}{dt} = -bxy \right\} \quad (64)$$

with state equation

$$b(x_0 - x) = a(y_0 - y), \quad (65)$$

hence the so-called “linear law.” Considering the effect of target, vice firer strength, on both sides yields

$$\left\{ \frac{dx}{dt} = -ax, \frac{dy}{dt} = -by \right\} \quad (66)$$

with state equation

$$b \log_e \frac{x_0}{x} = a \log_e \frac{y_0}{y}, \quad (67)$$

hence the “logarithmic law.”

Considering firer strength on one side and both firer and target strength on the other,

$$\left\{ \frac{dx}{dt} = -ay, \frac{dy}{dt} = -bxy \right\} \quad (68)$$

produces the mixed law

$$\frac{b}{2}(x_0^2 - x^2) = a(y_0 - y). \quad (69)$$

As alluded to earlier, such extensions can go on to the limits of the analyst’s abilities to justify the formulation and deal with the generally complicated and intractable solutions. As another illustration of this, the system

$$\left\{ \frac{dx}{dt} = -ay - \beta x, \frac{dy}{dt} = -bx - \alpha y \right\} \quad (70)$$

could arise by combining square law interactions among forces with fire support that is not attrited, or alternatively by considering square law interactions among forces having operational losses (48).

Hartley claims the Lanchester square law and linear laws do not provide good models of combat attrition as evidenced by historical data, but that a particular homogeneous linear-logarithmic law provides a good approximation to historical data (6). Another possibility we considered, one having some additional theoretical basis, is that of using a Helmbold-type system; for instance,

$$\left\{ \frac{dx}{dt} = -a \left(\frac{x}{y} \right) \exp(1 - \omega) y, \frac{dy}{dt} = -b \left(\frac{y}{x} \right) \exp(1 - \omega) x \right\} \quad (49). \quad (71)$$

9.3 Next Formulation

By way of introduction to this subsection, we note the extensions to the initial model cited earlier (even significant changes to the attrition representation), although improvements in an abstract sense, are obviously not sufficient for combat situations having any basis in reality. For instance, consider positional effect on attrition. Let us say as a first approximation that lethality follows an inverse-square scheme with distance, based on presented area considerations of sensing and weapon-delivery error if nothing else. Again, as a tactically realistic improvement, there should be a minimum and maximum range. Notions such as these are more or less readily incorporated into system equations.

Ancker finds combat theory and modeling basically deficient (50). However, he asserts that two statements can be considered as axioms: a firefight is a terminating stochastic target attrition process based on a discrete state space with a continuous time parameter, and all combat is a hierarchical network of firefights. He also notes that stochastic duel theory considers “microscopic” features of combat (e.g., time between firings, cover) as opposed to Lanchester’s aggregated effects. He also notes, however, that it has been shown possible to proceed from relatively easily measured microscopic parameters (51). Invoking theoretical relationships then helps lend veracity to macroscopic developments. We hope to investigate such linking of granularities in further mathematical analyses by follow-on collaboration with another ARL researcher who is extending such work into the realm of survivability analysis (52).

Battle command is harder to model than attrition, partly because of the immensely complicated nature of the interactions among entities. We realize this sort of work has been debated for years in military operations research circles. For example, there are Spradlin-type extensions (discussed in the next subsection) to more realistically model the behavior of an intelligent army. Such an entity has radios to relay information instantly, can react to the presence of opposing forces before actual contact, learn from mistakes, and not become stuck in undesirable equilibria.

Such items can be added incrementally as improvements to the modeling process. Moreover, our basic decision aiding methodology lends itself to essentially any of the formulations considered by analysts, given properly tailored mathematical algorithms.

However, we considered how to modify our initial formulation to reflect the fundamental functions of a military entity: shoot, move, and communicate. The intent was not to definitize any particular formulation, but rather to set forth a system that lends itself to incremental development in portraying more dimensions of combat beyond simple attrition. This serves two purposes: (1) enable an extension of the analytical/display aspects of the evolving decision aid prototype and, more importantly, (2) produce a dynamic system that can exhibit inherently chaotic behavior.

One approach was to add onto the set of “shoot” attrition equations

$$\left\{ \frac{dx_s}{dt} = -ay_s, \frac{dy_s}{dt} = -bx_s \right\} \quad (72)$$

a “move (or position)” set, say of the form

$$\left\{ \frac{dx_p}{dt} = \alpha(k - y_s), \frac{dy_p}{dt} = -\beta(k - x_s) \right\}, \quad (73)$$

and a “communicate” set, say of the form

$$\left\{ \frac{dx_c}{dt} = Ax_s, \frac{dy_c}{dt} = By_s \right\}. \quad (74)$$

Assume further that Blue is attacking on the move and Red is defending from a stationary position. We make this particular assumption basically in order to leverage the simulation results of the DISAF experimentation being done for our datamining research. However, there is some rationale in keeping the formulation relatively simple at first, in a spirit of iteratively increasing the sophistication as problems are encountered and solved. As noted in subsection 8.3, even apparently simple systems of equations can produce unexpectedly complicated behavior.

In this context, we considered qualitative descriptions: at fixed distance and communication ability, rate of destruction is proportional to strength; at fixed strengths and communication, rate of destruction \propto distance⁻²; at fixed strength and distance, rate of destruction \propto communication of opponent; at fixed strength and distance, rate of destruction \propto (own communication)⁻¹; at fixed strengths, movement \propto own communication. These sorts of criteria can be sharpened with experience and generalized into other tactical considerations. Such systems can be much more interdependent as attempts to increase realism are made. For example, we need to develop another system of linked equations in which sense/perceive and decide are modeled as well.

Modification of attrition to include effects of communications is problematic. It could be argued that in the context of our modeling perfect communication defaults to the basic square law. If no communication among entities exists, one could say that no effective fire, and hence no attrition of the enemy, is possible; but that seems unreasonable, even as a first cut, and one should probably consider a minimum effect even in the absence of communications. However, perhaps it would be better to consider $C \in [0,1]$ with a multiplicative effect on attrition of the form C^2 , where the parabolic effect is based on an n^2 network centric warfare (NCW) argument concerning the number of nodes in a network (53).

By considering such arguments, we developed systems of the following form that we intend to analyze using the software previously described.

Shoot:

$$\begin{aligned}\frac{dS_B}{dt} &= -aS_R(C - P_B)C_R \\ \frac{dS_R}{dt} &= -bS_B(C - P_B)C_B\end{aligned}\tag{75}$$

Move:

$$\begin{aligned}\frac{dP_B}{dt} &= \alpha\left(\frac{S_B}{S_R} - d\right) \\ \frac{dP_R}{dt} &= -\beta\left(\frac{S_R}{S_B} - f\right)\end{aligned}\tag{76}$$

Communicate:

$$\begin{aligned}\frac{dC_B}{dt} &= AS_B^2 \\ \frac{dC_R}{dt} &= DS_R^2\end{aligned}\tag{77}$$

These are again notional formulations, where S_B represents Blue strength, P_B Blue position, C_B Blue communication ability; and analogously for Red. In any event, the thrust of the investigation is to describe mutual qualitative influences, using given extremal input values yielding reasonable outputs.

Another approach was precipitated by considering the positive-coefficient system

$$\{ \dot{x} = (-\alpha + by)x = -\alpha x + bxy, \dot{y} = (\beta - cx)y = \beta y - cxy \},\tag{78}$$

a model developed by Lotka and Volterra for interaction among a predator population x and a prey population y . The linear terms $-\alpha x$ and βy model decay of predator and growth of prey in the absence of the other population. The quadratics bxy and $-cxy$ model the effects of

interaction, where the number of interactions is assumed jointly proportional to the population of each species: the “mass-action” coefficients b and c measure the predator’s efficiency in converting prey into reproduction and the probability that an interaction kills one prey (13). This is an interesting model to consider as a partial basis for the next formulation, not only because of the analogies between the natural situation and combat, but because it is a system of nonlinear DEs with closed form solution. Other such models can be developed for situations such as competition for resources, cooperation, harvesting, overcrowding, and predator satiation, again with parallels in the realm of military conflict.

Further, by setting $\dot{x} = \dot{y} = 0$ in the Lotka-Volterra system we find equilibria $(0,0)$ and $(\beta/c, \alpha/b)$. Borrelli and Coleman show that the orbits are closed and the solutions periodic; the component curves show predator peaks lag prey peaks, the period of an orbit increases with distance from the equilibrium point, and the orbits go clockwise (13). It is readily apparent that these kinds of analytical results, based on combat information available in the modern battlespace, could be quite useful to the commander.

9.4 Modeling Continuous Forces

We would like to consider methods for dealing with battles in which the combatants are distributed in space according to density functions. For example, in a general formulation each differential element of force would engage the enemy according to a kill rate function (e.g., of distance to points within the enemy mass) and would be killed at a rate comprising some kind of summation over the enemy elements. Advancement of one or both of the forces would be a function of ongoing combat as well as of mission/strategy, so computation of the vector field would generally be difficult.

In a sense, the traditional Lanchester approach is analogous to Lagrangian fluid dynamics. However, for modeling the distribution and movement of forces, Eulerian equations are arguably more appropriate. In particular, attrition of forces interacting as a result of movement of the forward edge of the battle area lends itself to use of partial DEs, which can naturally handle densities and geometry. In particular, such formulations may permit analysis of interpenetration of forces and “unusual” battles that may occur in future scenarios. However, they are often mathematically intractable.

Spradlin and Fields have augmented the discrete modeling of the individual soldier with larger forces represented as continuous density distributions (54). Then movements and attrition can be modeled via variations on reaction-diffusion equations in two spatial variables and time. We need to examine this notion further, building on earlier work by ORNL researchers, in particular Azmy and Protopopescu (37). In guiding troop movement, many factors must be included (attrition, terrain, mission, visibility, obstacles, etc.), so computation of the vector field is generally difficult. This work has been pursued for some time, but for greater realism improvements to the method are needed, such as reaction prior to contact, instantaneous communications, and learning from mistakes. Certain aspects of controlling the battle, with or

without such formulations, were set forth in subsection 8.6. Perhaps we can eventually investigate application of Cauchy analysis to battle execution by means of controlling its boundary.

9.5 Heuristics and Estimation

A related effort would be investigation of other methodologies for developing numerical values for the “Lanchester” coefficients. Of course, the best situation obtains when the systems can be validated by actual combat figures. Traditionally, this is quite difficult, but we are confident that the not-too-distant future will bring the ability to obtain reasonable near-real-time values for many of the attributes, at least for Blue. Such empirical population of the abstract formulation might be based on historical data or records of field experiments. Also, problems of perception and measurement noise (associated with the “fog of war”) arise both in the formulation of equations and determination of coefficient values.

Dupuy made valiant attempts to use historical data in modeling combat (55). He lists 73 “variable effects factors:” weapons effects, terrain factors, weather factors, seasonal factors, air superiority, posture, mobility effects, tactical air effects, other combat factors, and intangible aspects. He also considers whether they are calculable, sometimes calculable, probably calculable but not yet, and probably individually incalculable. In the last category are morale, leadership, as well as (somewhat surprisingly, in our opinion) intelligence and technology. We further note that his work did not involve aspects of NCW or information operations, at least in any meaningful sense.

One issue that emerged during the investigations into parameter estimation was that of applicability. This is not the same as the notion discussed elsewhere as to the practicality of applying these sorts of mathematical analyses in general to combat. Rather, it involves considerations of, for example, the numbers of troops required for reasonableness in the estimated solutions. We found that although the basic approach to estimating attrition rates for opposing forces seems to work in analyzing cases of conventional combat, there seems to be a problem with low attrition values, as in the case of the Iraq war for allied forces. We are looking at combinations of smoothing techniques and schemes for picking the attrition estimates of attrition to determine the best method to be used in the majority of cases. It may prove necessary that certain situations require special treatment, a more refined approach. We hope to develop this further in the follow-on efforts.

Other plans for work in parameter estimation include a mixture of types of improvement. For instance, it is vital that non-constant reporting intervals be addressed, as well as the fact that the commander generally gets Blue and Red updates at different times. Although such items can be handled in a sense via interpolation schemes, it is important to consider the tactical reality of sporadic updates and estimates of various degrees of certainty. Dealing with bands of uncertainty and their effects on parameter estimation and replanning is a ripe research area. More sophisticated approaches to dealing with parameter combinations are being considered; for

instance, “running averages” of Blue and Red kill rates and force levels. Code optimization techniques to reduce calculation time are always a factor in our work.

Further model formulations can be inspired by Dunnigan’s twelve rules of thumb, “historical outcomes that consistently repeat themselves (56).” These are exemplified by (1) combat causes losses of 1–5% casualties per day per division, (2) combat vehicles are lost at 5–10 times the rate of personnel, and (3) an attacker needs at least three times (varying with size of forces) the combat strength to overcome a defender.

A qualitatively different result of continuation of the project could be a “library” of scenario runs categorized in many dimensions. Utilizing National Training Center exercise data as preserved in the Automated Historical Archives System has been cited as a possibility in this regard, as has the work of Dupuy (55) and various case-based reasoning researchers such as Thompson (57). However, the internal ARL work with OneSAF/DISAF lends itself more directly to this sort of ontology due to the control and consistency of the data collection. We were concerned early on about possible difficulties in handling asymmetric warfare. We knew that guerrilla formulations might have to be used for the model if the technique is to be accepted for Future Force missions. However, this surfaced somewhat unexpectedly in the form of special requirements needed for parameter estimation in casualty situations like those encountered in the Gulf Wars. Future research will extend the techniques used for estimation; and the fact that modern solvers can handle most systems of equations should alleviate certain technical problems with the conflict formulation.

9.6 Other Applications

We intend to use two versions of a single combat simulation program to develop techniques for transferring “real-world” data and simulation results back and forth among executing systems. For example, process A (using as input a certain COA) would be “monitored” to provide “actual execution” battlefield information. This information would be used to simulate portions of the battle in process B, the results of which would be fed back into the “real” battle. We can thereby explore algorithms for determining divergence of combat from the fight and for expeditiously developing modifications, including fragmentary orders, to the plan.

We intend to tie execution monitoring into OneSAF initially and then other simulations such as the Joint Military Art of Command Environment. We have developed sets of files recording the execution of a number of runs of two types of scenarios: a conventional tank battle and an urban assault. These files contain extremely detailed time-stamped information about the unfolding battles (5, 58). We have developed programming techniques for converting such information into force level updates, and it will be relatively straightforward to utilize these files as drivers to the prototype. As precursors to such an exercise we intend to try simulation-based approaches of a different nature; in particular, we will monitor a OneSAF run. If it is determined that the typical trajectory of the scenario is not like the square law, we can develop a simpler simulation where

the “real” values extracted for input to the decision aid are in fact developed from a square law situation, albeit one unknown to the user of the aid.

An interesting project would be to consider an historical battle such as Iwo Jima or 73 Easting and attempt to improve the outcome via analysis of the dynamics, perhaps by monitoring casualties and shifting allocation of firepower. Another eventual effort could be to use the prototype decision aid during an exercise such as Prairie Warrior and see whether the command staff found the monitoring and trajectory change suggestions of any real use in modifying their tactics or force composition during the play.

We note that there is nothing special about the orientation of this decision aid work toward the Blue commander, other than the fact that he is the intended user and that the information available to the friendly side is generally more accurate. We could use a tool similar to the prototype for analyzing the Red situation as well; that is, model-based planning and replanning lends itself to applications in intelligence and wargaming as well, due to certain symmetries in threat assessment. A related extension involves consideration of the postulated Red commander’s desires and his ability to reach that end state in analyzing the Blue situation. It is hoped we can leverage pioneering work in multisided game theoretical applications to combat that is being done at ARL (59).

In a spirit of sketching out future work on other sorts of decision aiding using the paradigm of model-based execution monitoring, we have developed vignettes of different kinds of planning and replanning. Again, these utilize Lanchester modeling as a basis, but the ideas are extensible to other formulations. The intent is to inspire researchers to “think outside the box” about how the commander may be helped by computer-based aids. It may turn out that these sorts of mathematical abstractions are in fact too far removed from tactical (especially, small unit) reality to be immediately useful to the commander in a particular mission. However, it is still a challenge to prove this rigorously; and the work may still find application to different sorts of analysis, including drivers for artificially intelligent assistants.

One aspect (examined briefly in section 7) along these lines is that of assessing the value of Blue troop reinforcement at a given time. This certainly is not a novel consideration, although the idea here would be more one of optimizing the time and amount. At the least the commander should be able to readily ascertain the value of reinforcement contingencies under consideration with regard to changing the flow of battle.

Another larger area of study is that of expanded mathematical analysis of the value of NII. Put somewhat facetiously, what good is battle command? What does accurate (in the sense of valid intelligence, good communications, logical planning, etc.) command and control buy the planner and executor with regard to achieving desired outcome and facilitating contingency planning? As the DoD C4ISR Cooperative Research Program (CCRP) has pointed out (60), these notions have been studied for years, yet the answer is somewhat elusive. On the other hand, the CCRP notes, “NCW provides opportunities to improve C2 and execution ... because (1) decision entities or

C2 elements will be more knowledgeable; (2) actor entities will be more knowledgeable; (3) actor and decision entities will be better connected; (4) sensor entities will be more responsible; and (5) the footprint of all entities will be much smaller” (53).

Coe and Dockery set forth some sketches of experimentation in partial DEs, stochastic DEs, Q-analysis, chaotic dynamics, catastrophe theory, cellular automata, fuzzy sets, fractals, and measure of effectiveness definition (61). Woodcock and Dockery also note the development of a program of advanced military analysis based on catastrophe theory, singularity theory, dissipative structures, reaction-diffusion systems, etc., which is leading to a combination of modeling and decision aiding in an “electronic workbench” (62). Such work must be followed up.

We hope to be able to tie some of these investigations into the work Liao has begun on time series analysis of our combat data in understanding battle states (10). In particular, we hope to investigate the possibility of identifying the existence and fractal dimensionality of an attractor for a given time series or showing that the system has an irreducible stochastic nature. Moreover, it would be desirable to determine the fractal dimension of the attractor (which provides information about the system’s predictability and sensitivity to initial conditions) and the minimal dimensionality of its phase space (which yields the number of variables that must be used in describing the dynamics). Nicolis and Prigogine suggest an algorithmic approach to such investigations that we hope to pursue (33). In addition to the other types of conflict parameters, we will consider the notion of the energy or potential of the battlespace, perhaps building on our earlier work on calculating intensity (5). Again, all of these are potentially of use to the commander given a meaningful link between model and reality; with the proper software, we look forward to analyzing these aspects of models of tactical systems.

10. Conclusion

10.1 Summary

By way of introduction to this conclusion, we note that, although we have presented several sections dealing with apparently disparate topics, they do in fact comprise an integrated “course of action” for continuing the work presented up to this point. The expositions dealt primarily with sketches of algorithms for, and interfaces to, a prototype decision aid. We considered several aspects of the analytical theory of dynamic systems, in particular mathematical chaos, stability, and control. Miscellaneous discussions involved “mechanical” follow-on work, such as simulated execution monitoring, parameter estimation, and other application-specific items.

The intent was to build in reasonable stages toward both theory and prototype: define approaches to major technical challenges, develop initial algorithms, begin graphical prototype, prototype improvements for monitoring of simulated conflict, and begin considerations of control theory

applications based on emerging results. We accomplished some, though not yet all, of what we set out to do in this project. We performed background research into approaches to various technical challenges. We designed an algorithmic solution to a prototype trajectory decision aiding system and explored some complexities of navigating parameter trade-off spaces. We did some detailed studies of strength/attrition parameter estimation and calibration techniques. We investigated software packages for solving differential systems and for visualizing and interacting with decision aid data. We examined methods for utilizing the results of OneSAF/DISAF executions. We set the groundwork for formulation of models to produce systems exhibiting inherently chaotic behavior and for studying qualitative aspects of phase plane trajectories.

As a result of the research, design, and software engineering begun in this project, we have the intellectual material for several in-depth expositions. These include the details of the parameter estimation investigations, applications of chaos theory to the study of military operations, and several spin-offs concerning COA theory, in particular the notion of plan divergence. We have more work to do with regard to iterative function theory and control theory applied to combat modeling, graphical prototyping, improving combat modeling via consideration of nonlinear dynamics, and approaches to applying stability analyses in combat monitoring.

10.2 Impact

It is probable that nonlinear analytic techniques will soon be considered as invaluable to commanders. We have seen that time series analysis can permit near-term prediction, attractors can indicate probabilities of outcomes, and Lyapunov analysis can help assess relative predictability of option results.

This work, although obviously still in a basic research stage, does have transition potential. The work is intended to lead to portions of an integrated system to dynamically link automated plan generation and analysis with execution monitoring. Careful tuning of software to specific missions could improve the quality and timeliness of current planning programs. This research could also yield improved complexity for large simulations. Perhaps sensitivity analyses may show the relative importance of parts of the model, even allowing for reduction of scope. In any event, it would be useful to investigate situations and parameters for which greater detail may not necessarily be better for the analysis.

Extensions of this effort can augment and improve ARL's ongoing successful research program toward methodologies to develop and evaluate COAs and to improve wargaming. Lessons learned may lead to extended COA measures of effectiveness. This work has potential for improving military command and control and analogous civilian systems. Scientific evaluations of plan attributes are of research value to the U.S. Army Communications—Electronics Research, Development, and Engineering Center and the Army Battle Command Battle Laboratory, and will benefit the soldier by revealing more numerous operational options during

planning and replanning. Other beneficiaries could include any activity requiring rapid evaluation of operational plan status, in particular emergency responders.

10.3 Issues

Several programmatic challenges arose over the course of this project. As mentioned in section 6, apparent software security issues resulted in a deviation from the plan; they must be addressed to allow substantial future research and development. Personnel issues and other duties negatively impacted concentrated effort throughout the period of performance. These detours did, however, produce an unexpected graphics development spin-off, with potential for a possible GUI patent.

We share the concern expressed in some circles over the ability of abstract systems to model combat, particularly military operations on urban terrain, in sufficient detail as to be generally tactically useful. Moreover, we believe discrete dynamics will play a larger part in analyses of small unit actions. However, we maintain that the utility of such techniques will be proven to the soldier in the forms of bounding parameters, portraying worst cases, and assessing possibilities. It is our vision that when the future force is operational, the Army will be able to effectively and efficiently utilize the enormous amounts of data provided by Future Combat Systems.

10.4 Final Observations

We trust the reader has been given a flavor of what work in the realm of “A Model-Based Approach to Battle Execution Monitoring” might involve over the long run. This report is merely an introduction to the possibilities for such decision aiding science and technology. Although we have just scratched the surface in developing a simple prototype, we have touched on disparate important aspects: the model, the mathematics, the display, and the dynamic analysis. This project precipitated more work on aspects of the theory of COA analysis and battle execution monitoring for replanning. More discussion of the development of these notions is planned for subsequent reports. We trust this work will be found worthy of continuance, because the true research sketched out by this framework is just beginning.

Collaborations in the form of exchanges and discussions are being pursued with several individuals in government and academia. For example, Dr. Mary Anne Fields of the ARL Weapons and Materials Research Directorate and Dr. Greg Spradlin of the U.S. Military Academy Mathematical Sciences Department are developing a battlefield model using reaction-diffusion equations (54). Jeffrey Smith of the ARL Survivability/Lethality Analysis Directorate is developing a method for analyzing survivability in the context of a one-on-one engagement (52). James Thompson of Rice University is exploring applications of Lanchester laws to modular wargaming (63). We plan to share progress with: the Naval Postgraduate School Operations Research Department, a center for Lanchester research; the Ohio State University on Multicriteria Decision Making; and the National Simulation Center on Command Decision

Modeling. We intend to collaborate further with several of these groups as our work gels into a vehicle for more in-depth research.

This project allowed us the opportunity to examine some novel issues, and we look forward to contributing further by extending our research into realms of deception analysis and understanding the value of information. Analyses of nonlinear dynamics, combined with the powerful techniques being developed by ARL researchers for combat simulation data mining, should result in better understanding of battle “predictors” and much improved battlespace decision support.

11. References

1. Lanchester, F. *Aircraft in Warfare: The Dawn of the Fourth Arm*; Lanchester Press: Sunnyvale, CA, 1916; pp 53–54.
2. FM 101-5-1; *Operational Terms and Graphics*; Department of the Army: Headquarters, Washington, DC, 1997.
3. Kaste, R.; Bodt, B.; Forester, J.; Hansen, C.; Heilman, E.; Korjack, T.; Thomas, M.; Wells, H. *Research in Modeling and Simulation for C2*; ARL-SR-97; U.S. Army Research Laboratory: Aberdeen Proving Ground, MD, 2000.
4. Barnes, M.; Hillis, D. *Dynamic Visualization of Co-Evolving Adversaries in Small-Scale Contingencies*; Compendium of FY02 DRI Final Reports, 2003; U.S. Army Research Laboratory: Adelphi, MD, 2003, 43–57.
5. O'May, J.; Heilman, E.; Bodt, B.; Forester, J. OneSAF Killer/Victim Scoreboard Capability for C2 Experimentation. *Proceedings of 2002 C2 Research and Technology Symposium*, Naval Postgraduate School, Monterey, CA, 2002.
6. Anderson, L.; Miercort, F. *On Weapons Scores and Force Strengths*; Warfare Modeling, MORS; Wiley: New York, 1995.
7. Hansen, C.; Heilman, E.; Kaste, R.; O'May, J. *Battlespace Terrain Ownership: A New Situation Awareness Tool*; Army Science Conference paper, in process.
8. Heilman, E.; O'May, J. *OneSAF Killer/Victim Scoreboard Capability*; ARL-TR-2829; U.S. Army Research Laboratory: Aberdeen Proving Ground, MD, 2002.
9. *Mathematics of Military Action, Operations, and Systems*; DA Pamphlet No. 70-5; Headquarters, Department of the Army, 1968.
10. Liao, T.; Bolt, B.; Forester, J.; Hansen, C.; Heilman, E.; Kaste, R.; O'May, J. *Understanding and Projecting Battle States*; 25th Army Science Conference, Orlando, FL, 2002.
11. Caldwell, B.; Hartman, J.; Parry, S.; Washburn, A.; Youngren, M. *Aggregated Combat Models*; Naval Postgraduate School, Monterey, CA, 2000.
12. Taylor, J. *Lanchester-Type Models of Warfare*; Naval Postgraduate School: Operations Research Department, Monterey, CA, 1985.
13. Borrelli, R.; Coleman, C. *Differential Equations: A Modeling Perspective*; Wiley: New York, 1996; pp 46, 362–6, 414–5, 323, 323–4.

14. Institute for Human Machine Cognition Website. <http://www.ihmc.us/research/HumanCenteredVisualization/> (accessed 2003).
15. U.S. Army Research Laboratory Website. <http://www.arl.army.mil/alliances/adaoview.htm> (accessed 2003).
16. Press, W.; Flannery, B.; Teukolsky, S.; Vetterling, W. *Numerical Recipes: The Art of Scientific Computing*; Cambridge University Press: Cambridge, UK, 1986.
17. Josephson, J.; Chandrasekaran, B.; Carroll, M.; Iyer, N.; Wasacz, B.; Rizzani, G.; Li, Q.; Erb, D. *An Architecture for Exploring Large Design Spaces*; AAAI-98, Madison, WI, 1998.
18. Kaste, R.; O'May, J.; Heilman, E.; Chandrasekaran, B.; Josephson, J. From Simulation to Insights: Experiments in the Use of a Multi-criterial Viewer to Develop Understanding of the COA Space. *ARL CTA Proceedings*, College Park, MD, 2003.
19. Center for the Representation of Multi-Dimensional Information Home Page. <http://www.cromdi.utah.edu> (accessed 2003).
20. Hilborn, R. *Chaos and Nonlinear Dynamics: An Introduction for Scientists and Engineers*; Oxford University Press: New York, 1994; pp 64–65.
21. Parker, T.; Chua, L. *Practical Numerical Algorithms for Chaotic Systems*; Springer-Verlag: New York, 1989.
22. Kocak, H. *Differential and Difference Equations through Computer Experiments*, 2nd Ed.; Springer-Verlag: New York, 1989.
23. Peitgen, H-O.; Jürgens, H.; Saupe, D. *Chaos and Fractals: New Frontiers of Science*; Springer-Verlag: New York, 1992.
24. Aguirregabiria, J. Dynamics Solver. <http://tp.lc.ehu.es/jma/ds/ds.html> (accessed 2003).
25. Van der Lee, J. JPlot. <http://graphs.sourceforge.net/index.html> (accessed 2003).
26. Knuth, D. *The Art of Computer Programming: Fundamental Algorithms*, 3rd Ed., Addison-Wesley: Boston, MA, 1997.
27. Anderson, L. *Antipotential Potential*; Institute for Defense Analyses Note N-845, 1979.
28. Cooper, P. Coalition Deaths Fewer Than in 1991. <http://www.cnn.com/2003/WORLD/meast/04/17/sprj.irq.casualties/index.html> (accessed 2003).
29. Nemytskii, V.; Stepanov, V. *Qualitative Theory of Differential Equations*; Dover Publications: New York, 1989.

30. Czerwinski, T. *Coping with the Bounds: Speculations on Nonlinearity in Military Affairs*; National Defense University, Washington, DC, 1998.
31. Weaver, W. Science and Complexity. *American Scientist* **1948**, 36, 536–544.
32. James, G. *Chaos Theory: The Essentials for Military Applications*; Naval War College, Newport, RI, 1996.
33. Nicolis, G.; Prigogine, I. *Exploring Complexity: An Introduction*; W. H. Freeman and Co.: New York, 1989.
34. Weisstein, E. “World of Mathematics.” <http://mathworld.wolfram.com> (accessed 2003).
35. Bradley, E. *Taming Chaotic Circuits*; MIT AI Lab TR 1388: Cambridge, MA, 1992.
36. Hayes, S. Communicating with Chaos: A Physical Theory for Communication via Controlled Symbolic Dynamics. Ph.D. Thesis, University of Maryland, College Park, MD, 1996.
37. Azmy, Y.; Protopopescu, V. *Two Dimensional Maps Generated by Competitive Systems*; ORNL TM-11026; Oak Ridge National Laboratory: Oak Ridge, TN, 1989.
38. Dewar, J.; Gillogly, J.; Juncosa, M. *Non-Monotonicity, Chaos, and Combat Models*; RAND: Santa Monica, CA, 1991.
39. Dockery, J.; Woodcock, A. *The Military Landscape*; Woodhead Publishing: Cambridge, UK, 1993.
40. Wiggins, S. *Introduction to Applied Nonlinear Dynamical Systems and Chaos*; Springer-Verlag: New York, 1990.
41. Wylie, C.; Barrett, L. *Advanced Engineering Mathematics*, 6/e; McGraw-Hill: New York, 1995.
42. Raglin, A. *Large Array of Opto-Electronic Feedback Circuits With Winner-Take-All Neural Network Dynamics for Image Processing Applications*; Compendium of FY03 DRI Final Reports; U.S. Army Research Laboratory: Adelphi, MD, 2004, pp 81–101.
43. Gilmore, R. *Catastrophe Theory for Scientists and Engineers*; Wiley: New York, 1981.
44. Liao, T.; et al. *Genetic Clustering of Time Series Data: An Application to Battle Simulations*; Louisiana State University: Baton Rouge, LA, in process.
45. Callahan, Jr., L.; Stuk, S. *Multivariable Control Theory for Lanchester Type Models*; Georgia Institute of Technology: Atlanta, GA, 1985.
46. Luenberger, D. *Introduction to Dynamic Systems: Theory, Models, & Applications*; Wiley: New York, 1979.

47. Brogan, W. *Modern Control Theory*; Quantum Publishers: New York, 1974.
48. Taylor, J. *Force-on-Force Attrition Modelling*; Operations Research Society of America: New York, 1980.
49. Helmbold, R. *Some Observations on the Use of Lanchester's Theory for Prediction*; Operations Research: Lincoln, RI, 1964.
50. Bracken, J.; Kress, M.; Rosenthal, R. *Warfare Modeling*; Wiley: New York, 1995.
51. Ancker, Jr., C. *One-on-One Stochastic Duels*; Operations Research Society of America: Arlington, VA, 1982.
52. Smith, J. *A Method for Analyzing Survivability in the Context of a One-on-One Engagement*; ARL-TR-2992; U.S. Army Research Laboratory: White Sands Missile Range, NM, 2003.
53. Alberts, D.; Garstka, J.; Stein, F. *Network Centric Warfare: Developing and Leveraging Information Superiority*, 2nd Ed.; C4ISR Cooperative Research Program: Washington, DC, 1999.
54. Spradlin, G. *Modeling Attrition in the Dynamic Two-Dimensional Battlefield*; Embry-Riddle Aeronautical University: Daytona Beach, FL, 2000.
55. Dupuy, T. *Numbers, Predictions, and War: The Use of History to Evaluate and Predict the Outcome of Armed Conflict*; HERO Books: Annandale, VA, 1985.
56. Dunnigan, J. *How to Make War: A Comprehensive Guide to Modern Warfare*; Quill: New York, 1988.
57. Thompson, J. *A Proposal for Computer Based Combat Simulation*; Rice University: Houston, TX, 2003.
58. O'May, J.; Heilman, E.; Bodt, B. *Data Mining Techniques Applied to Urban Terrain C2 Experimentation*; *C2 Research and Technology Symposium*; National Defense University: Washington, DC, 2003.
59. Hillis, D.; et al. *Collaborative Visualization Tools for COA in Anti-Terrorist Operations*. *ARL CTA Proceedings*, College Park, MD, 2003.
60. OASD-NII, Command and Control Research Program Website. <http://www.dodccrp.org> (accessed 2003).
61. Johnson, S.; Levis, A., Eds. *Science of Command and Control: Coping with Uncertainty*; AFCEA International Press: Fairfax, VA, 1988.
62. Woodcock, A.; Dockery, J. *Mathematical Catastrophes and Military Victories*; SIGNAL: Fairfax, VA, 1987.
63. Thompson, J. *Empirical Model Building*; Wiley: New York, 1989.

NO. OF
COPIES ORGANIZATION

1 DEFENSE TECHNICAL
(PDF INFORMATION CTR
ONLY) DTIC OCA
8725 JOHN J KINGMAN RD
STE 0944
FORT BELVOIR VA 22060-6218

1 US ARMY RSRCH DEV &
ENGRG CMD
SYSTEMS OF SYSTEMS
INTEGRATION
AMSRD SS T
6000 6TH ST STE 100
FORT BELVOIR VA 22060-5608

1 INST FOR ADVNCD TCHNLGY
THE UNIV OF TEXAS
AT AUSTIN
3925 W BRAKER LN STE 400
AUSTIN TX 78759-5316

1 US MILITARY ACADEMY
MATH SCI CTR EXCELLENCE
MADN MATH
THAYER HALL
WEST POINT NY 10996-1786

1 DIRECTOR
US ARMY RESEARCH LAB
IMNE AD IM DR
2800 POWDER MILL RD
ADELPHI MD 20783-1197

3 DIRECTOR
US ARMY RESEARCH LAB
AMSRD ARL CI OK TL
2800 POWDER MILL RD
ADELPHI MD 20783-1197

3 DIRECTOR
US ARMY RESEARCH LAB
AMSRD ARL CS IS T
2800 POWDER MILL RD
ADELPHI MD 20783-1197

NO. OF
COPIES ORGANIZATION

ABERDEEN PROVING GROUND

1 DIR USARL
AMSRD ARL CI OK TP (BLDG 4600)

NO. OF
COPIES ORGANIZATION

- 2 US ARMY RESEARCH OFFICE
AMSRD ARL RO MM
J LAVERY
D HISLOP
PO BOX 12211
RESEARCH TRIANGLE PARK NC
27709-2211
- 3 COMMANDER
US ARMY CERDEC
C2 DIRECTORATE
T NEWSOME
I MAYK
J SOOS
MYER CENTER
FT MONMOUTH NJ 07703-5630
- 3 DIR USARL
AMSRD ARL CI CS
G YERACE
J GOWENS
J GANTT
2800 POWDER MILL RD
ADELPHI MD 20783-1197
- 1 DIR USARL
AMSRD ARL SL EM
J SMITH
WSMR NM 88002-5513

ABERDEEN PROVING GROUND

- 16 DIR USARL
AMSRD ARL WM BA
A THOMPSON
AMSRD ARL CI CT
P JONES
C HANSEN (3 CPS)
R KASTE (10 CPS)
AMSRD ARL WM BF
M FIELDS

INTENTIONALLY LEFT BLANK.